2026/09/20, 2026 보안 기초 세미나

암호학과 네트워크 보안

- 16장 PGP와 S/MIME -

이 성 현(seonghyeon@pel.sejong.ac.kr)

세종대학교 프로토콜공학연구실

목 차

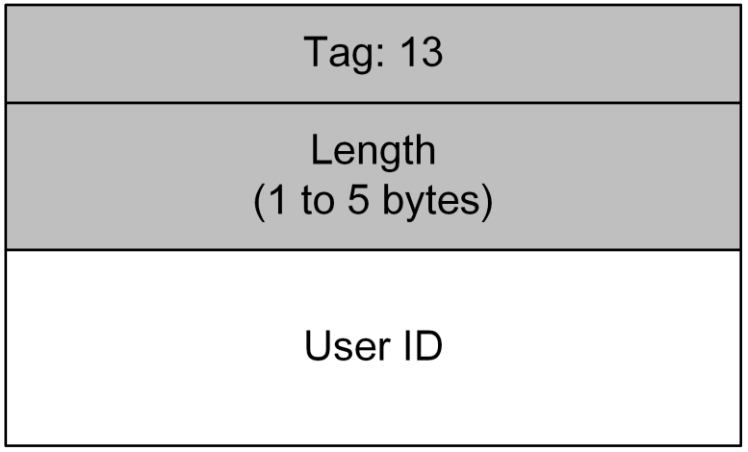
- 보충
- 개요
- PGP
- S/MIME

목 차

- 보충
- 개요
- PGP
- S/MIME

보충

- PGP (Pretty Good Privacy) (1/4)
- 패킷
  - User ID 패킷(User ID Packet)
    - User ID
      - 송신자의 User ID를 나타내며, UTF-8로 부호화된 문자열로 저장됨
      - 주로 전자우편 주소에 따른 사용자의 이름으로 구성됨



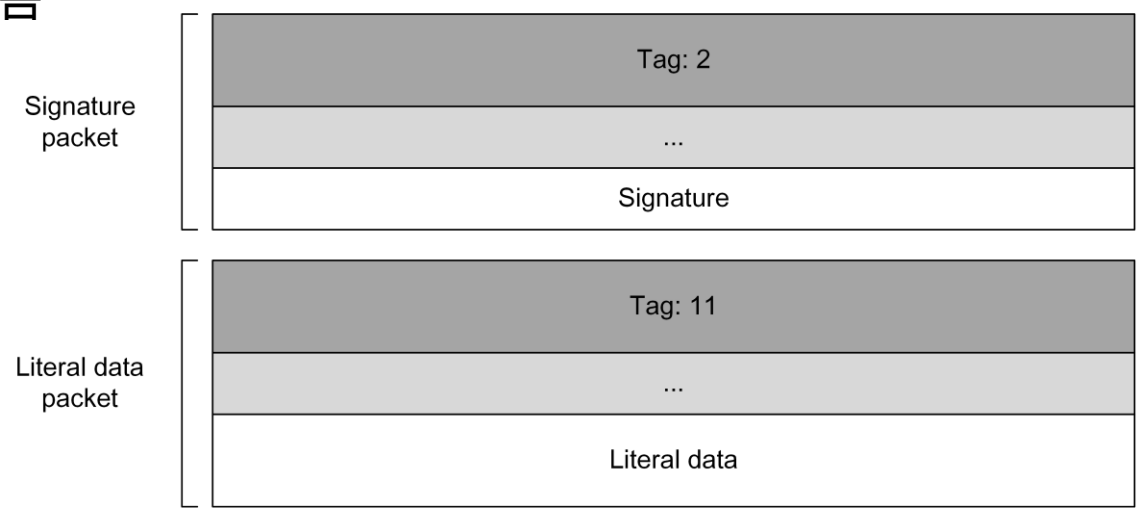
보충

- PGP (Pretty Good Privacy) (2/4)
  - 메시지 (1/3)
    - 암호화된 메시지(Encrypted Message)
      - 메시지를 암호화하여 보낼 때 사용함
      - Session Key Packet과 해당 패킷의 세션키로 암호화된 Encrypted Data Packet으로 구성될 수 있음
        - Encrypted Data Packet의 Encrypted data는 Compressed Data Packet으로 구성됨
        - Compressed Data Packet의 Compressed data는 Literal Data Packet으로 구성됨
        - 추가적인 패킷을 전송하고자 하는 경우, 새로운 메시지로 생성해야 함



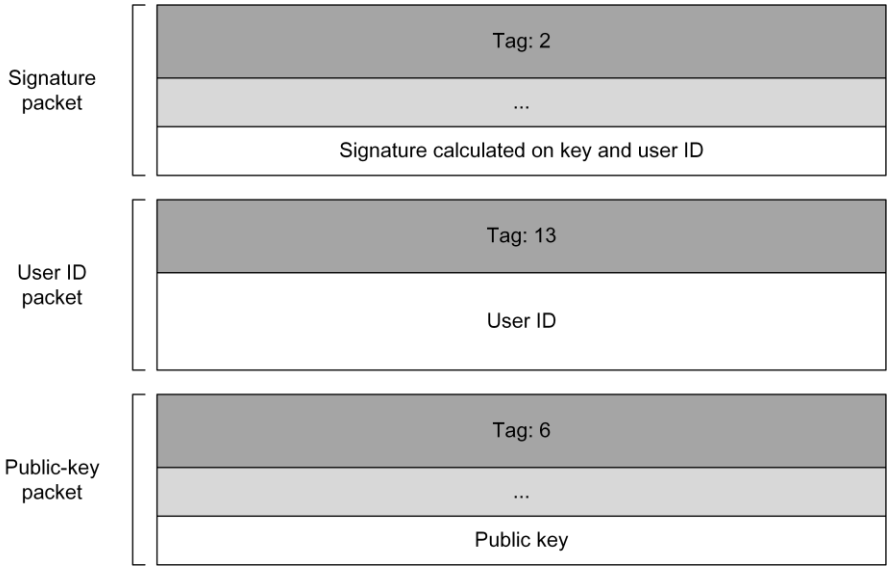
보충

- PGP (Pretty Good Privacy) (3/4)
  - 메시지 (2/3)
    - 서명된 메시지(Signed Message)
      - 메시지에 대한 서명을 보낼 때 사용함
      - Signature Packet과 Literal Data Packet으로 구성될 수 있음
        - Literal Data Packet은 필요에 따라 압축하여 Compressed Data Packet으로 만들거나, 압축한 후 암호화하여 Encrypted Data Packet으로 구성할 수 있음
        - 서명자가 여러 명인 경우, 서명자만큼의 Signature Packet을 추가할 수 있음



보충

- PGP (Pretty Good Privacy) (4/4)
  - 메시지 (3/3)
    - 인증서 메시지(Certificate Message)
      - 공개키가 해당 User ID를 가진 사용자의 공개키임을 서명으로 인증하여 보낼 때 사용함
      - Signature Packet, User ID Packet, Public-key Packet으로 구성될 수 있음
        - 인증서 구성에 따라 패킷이 추가될 수 있음



목 차

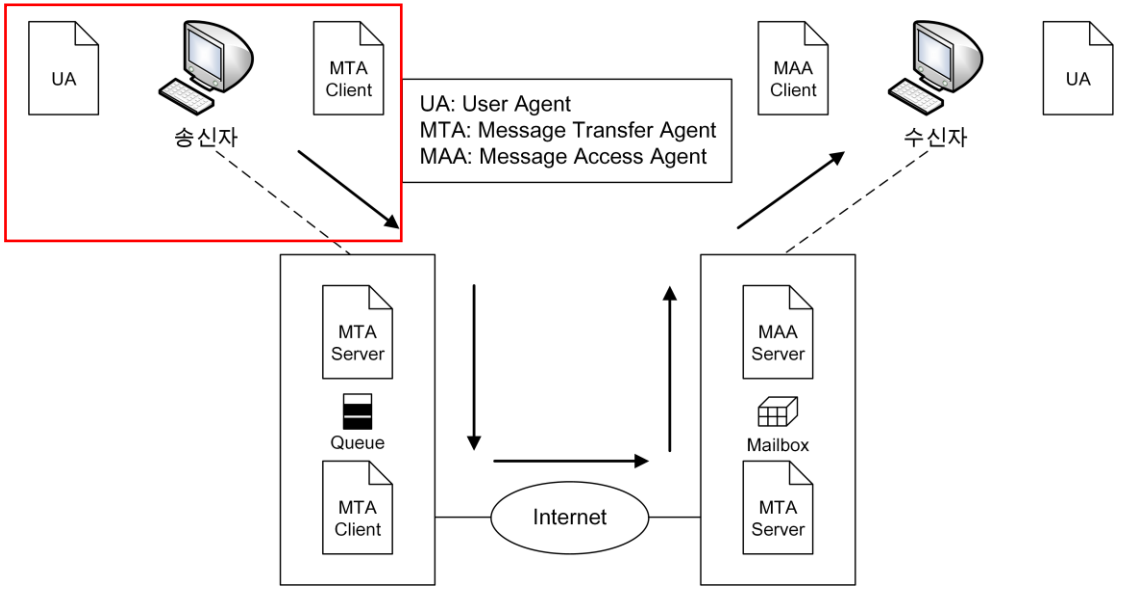
- 보충
- 개요
- PGP
- S/MIME

개요

- 전자우편 – 구조 (1/5)
  - 구성
    - UA (User Agent)
      - 사용자가 전자 우편을 작성, 열람, 관리할 때 사용하는 에이전트
      - 사용자 인터페이스 역할을 수행함
    - MTA (Message Transfer Agent)
      - 전자우편을 메일 서버 간 전송하는 에이전트
      - 클라이언트/서버 구조를 가지며, Push 방식으로 메일을 전송함
    - MAA (Message Access Agent)
      - 사용자가 메일 서버의 전자우편에 접근하도록 하는 에이전트
      - 클라이언트/서버 구조를 가지며, Pull 방식으로 메일에 접근함

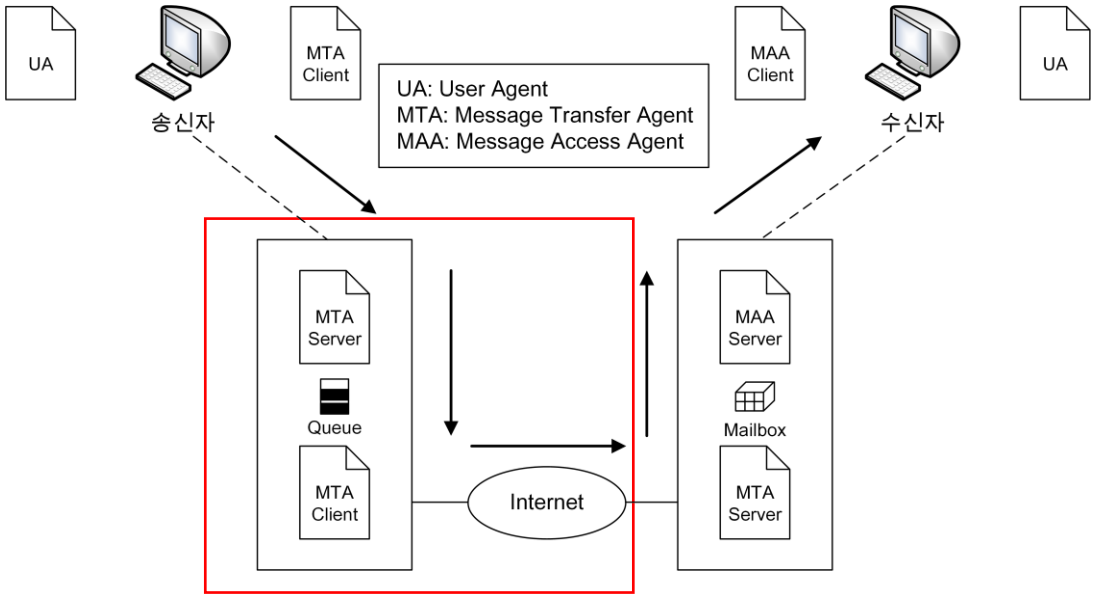
개요

- 전자우편 – 구조 (2/5)
  - 과정 (1/4)
    1. 송신자가 UA를 이용하여 전자우편을 작성함
    2. 송신자측 MTA 클라이언트가 UA로부터 전자우편을 전달받아 송신자측 메일서버의 MTA 서버로 전달함



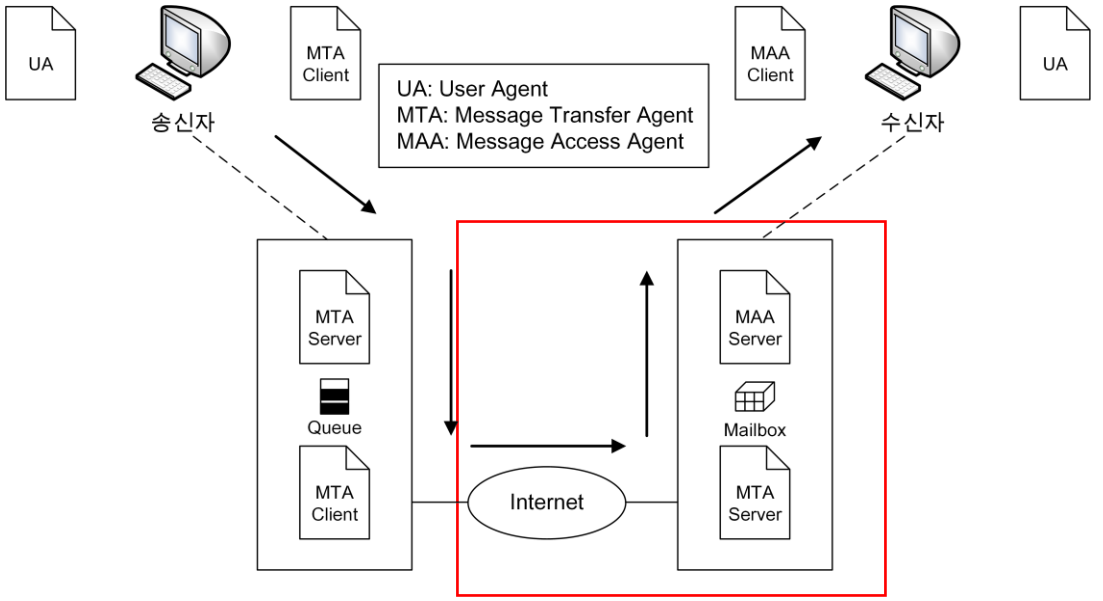
개요

- 전자우편 – 구조 (3/5)
  - 과정 (2/4)
    3. 송신자측 메일서버의 MTA 서버가 메시지를 큐에 저장함
    4. 송신자측 메일서버의 MTA 클라이언트가 큐에서 전자우편을 가져와 수신자측 메일서버로 전송함



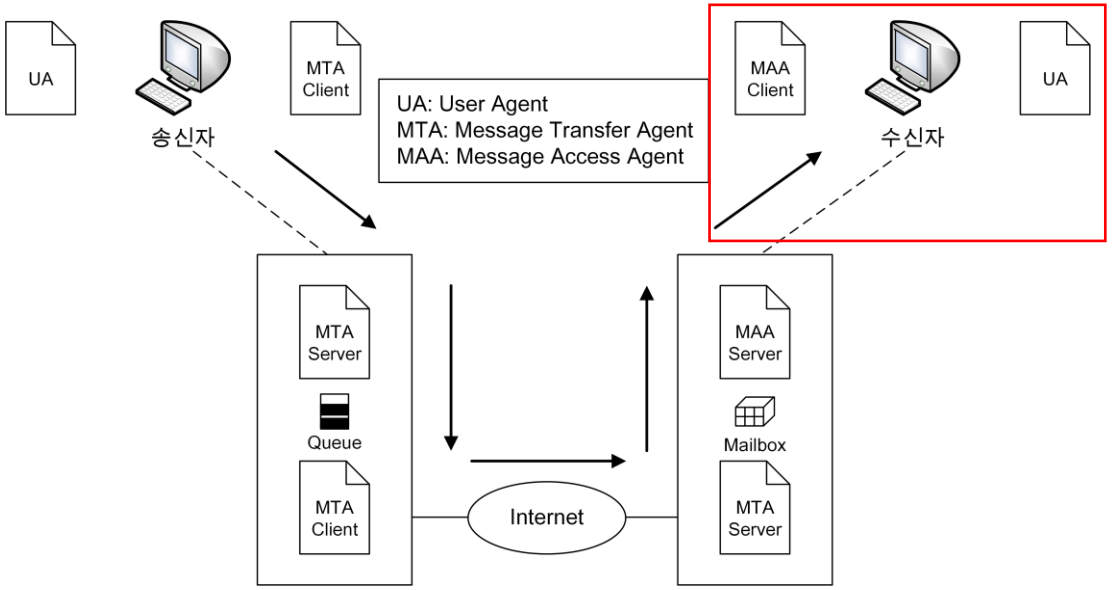
개요

- 전자우편 – 구조 (4/5)
  - 과정 (3/4)
    5. 수신자측 메일서버의 MTA 서버가 전자우편을 수신하여 수신자측 메일박스에 저장함
    6. 수신자측 MAA 클라이언트가 수신자측 메일서버의 MAA 서버에게 전자우편을 요청함



개요

- 전자우편 – 구조 (5/5)
  - 과정 (4/4)
    7. 수신자측 메일서버의 MAA 서버가 메일박스에서 전자우편을 가져와 수신자측 MAA 클라이언트에게 전달함
    8. 수신자가의 UA를 통해 전자우편을 열람함



개요

- 전자우편 – 보안 (1/2)
  - 개요
    - 송신자가 전자우편을 보내도 수신자는 해당 전자우편을 나중에 열람하거나 답장할 수 있음
    - 전자우편의 보안은 송신자와 수신자 간에 세션을 맺지 않아도 독립적으로 동작할 수 있어야 함
  - 암호 알고리즘
    - 세션을 맺지 않은 상태에서도 전자우편 보안을 위한 암호 알고리즘을 설정해야 함
      - 단일 알고리즘만 사용하도록 강제하는 경우
        - 모든 사용자가 동일한 암호 알고리즘을 사용하여 별도의 협상이 필요하지 않지만, 알고리즘 교체 시 전체 시스템에 영향을 미침
      - 사용 가능한 알고리즘 중에서 선택하도록 하는 경우
        - 전자우편을 위한 알고리즘 선택에 유연성이 생기게 되지만, 송신자는 사용한 알고리즘의 식별자를 반드시 포함해야 함

개요

- 전자우편 – 보안 (2/2)
  - 암호학적 비밀
    - 비대칭키 암호화는 길이가 긴 메시지의 암호화에 부적합 하므로 대칭키를 이용한 암호화가 적합함
    - 대칭키 암호화를 위한 비밀 설정은 일회용 비밀키를 수신자의 공개키로 암호화하여 메시지와 함께 보내 해결함
  - 인증서
    - 전자우편의 송수신 과정에서 송신자는 수신자의 공개키로 비밀키를 암호화 해야하며, 수신자는 송신자의 공개키로 서명을 검증해야 함
    - 공개키를 신뢰하기 위한 인증서는 사용하는 전자우편 보안 방식마다 다르게 운용됨

목 차

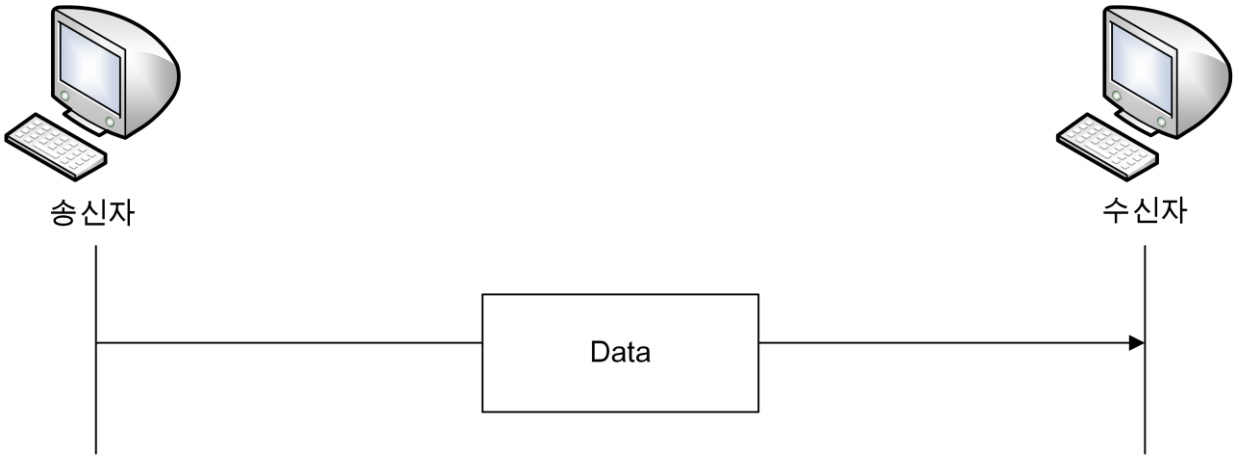
- 보충
- 개요
- PGP
- S/MIME

PGP

- 개요
  - PGP (Pretty Good Privacy)는 전자우편의 개인정보보호 (Privacy Protection), 무결성(Integrity), 인증(Authentication)을 제공하기 위해 필 짐머만(Phil Zimmermann)에 의해 고안됨
  - 안전한 전자우편 메시지를 생성하거나, 향후에 검색을 위해 파일을 안전하게 저장하는 데에 사용될 수 있음
- 제공 기능
  - 무결성
  - 기밀성
  - 압축
  - 부호변환
  - 단편화

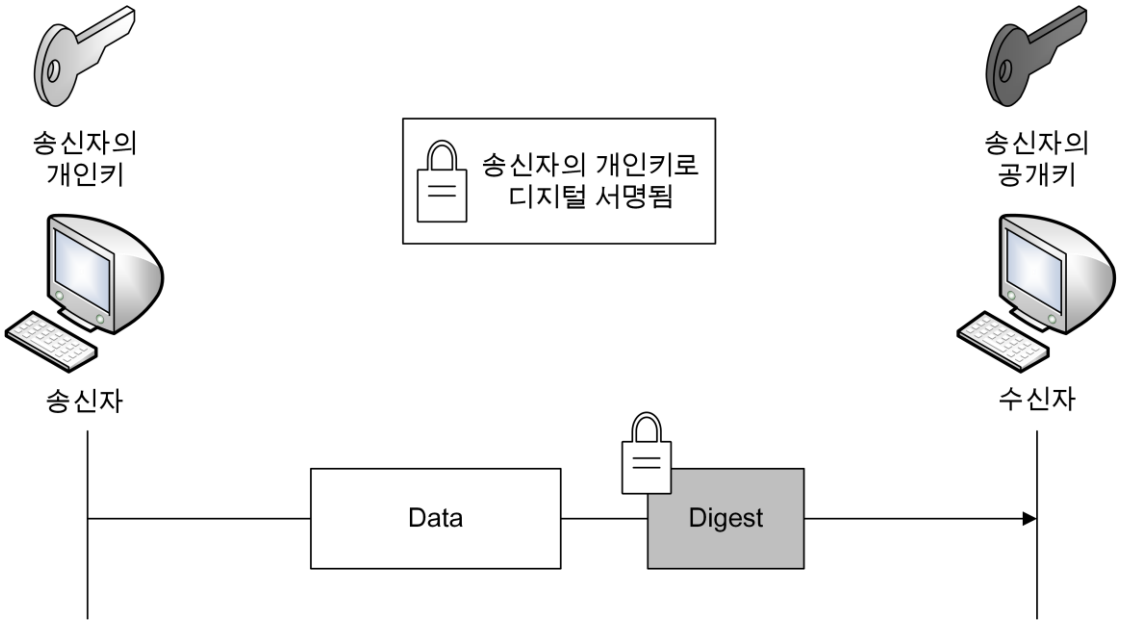
PGP

- 메시지 처리 방식 (1/5)
  - 평문
    - 전자우편 메시지를 평문으로 전송할 수 있음
    - 평문으로 전송하는 경우, 전송되는 전자우편의 무결성 또는 기밀성을 보장할 수 없음



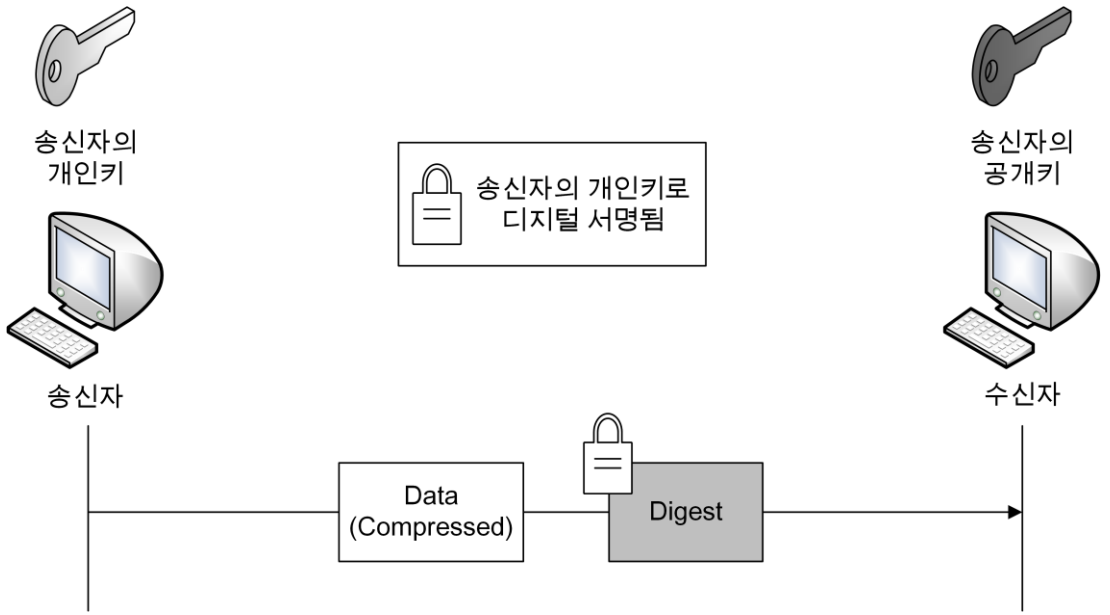
PGP

- 메시지 처리 방식 (2/5)
  - 무결성
    - 전자우편의 무결성을 위해 메시지에 대한 다이제스트(Digest)를 생성하고 해당 다이제스트를 자신의 개인키로 서명하여 전송할 수 있음
    - 수신자는 송신자의 공개키로 수신받은 서명을 검증함



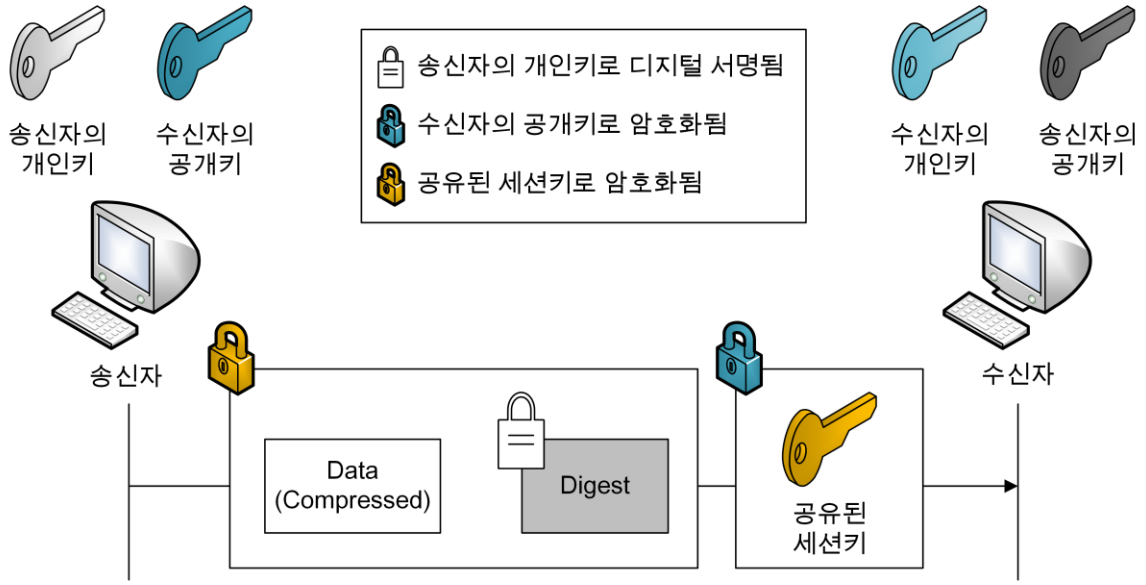
PGP

- 메시지 처리 방식 (3/5)
  - 압축
    - 패킷의 크기를 줄이고 트래픽에 대한 부담을 낮추기 위해 메시지를 압축할 수 있음
    - 송신자는 사용한 압축 알고리즘을 포함한 압축된 메시지를 전송함



PGP

- 메시지 처리 방식 (4/5)
  - 기밀성
    - 기밀성을 보장받기 위해 메시지와 서명을 세션키로 암호화하고 이를 수신자의 공개키로 암호화하여 전송할 수 있음
    - 수신자는 자신의 개인키로 세션키를 복호화하고 이를 이용하여 메시지와 서명을 복호화함



PGP

- 메시지 처리 방식 (5/5)
  - 부호 변환
    - PGP는 전자우편 시스템에서의 호환성을 위해, 전송하려는 메시지를 Radix-64 변환을 사용하여 ASCII 코드 값으로 변환하여 전송함

Radix-64:
2진 데이터를 64개의 ASCII 문자 집합을 이용해 텍스트 형태로 변환하는 인코딩 방식

  - 단편화
    - PGP는 상황에 따라 큰 메시지를 단편화하여 전송할 수 있음
    - 단편화를 하려는 경우, Radix-64 변환을 통해 ASCII 코드 값으로 변환된 값을 단편화 함

PGP

- 인증서(Certificate) (1/6)

X.509 인증서:
인증기관(CA, Certificate Authority)이 공개키와 소유자의 신원 정보를 결합하여 발급하는 디지털 인증서

  - 개요
    - X.509 인증서는 공개키에 대한 인증 주체가 인증기관(CA, Certificate Authority)이며, 인증서를 신뢰하기 위해서는 신뢰할 수 있는 CA가 발급했는지가 중요함
    - PGP 인증서는 공개키에 대한 인증 주체가 다른 PGP 사용자이며, 인증서를 누가 서명했고 그 사람을 얼마나 신뢰하고 있는지가 중요함
    - PGP의 완전한 동작은 소개자(Introducer)의 신뢰(Trust), 인증서 신뢰, 공개키의 적법성(Legitimacy)를 기반으로 함

PGP

- 인증서(Certificate) (2/6)
  - 소개자 신뢰등급
    - 소개자를 신뢰하는 정도에 등급(Level)을 부여할 수 있음
    - 주로 신뢰하지 않음(None), 부분적(Partial), 완전한(Full)로 나눌 수 있으며, 이러한 결정은 사용자가 내림
  - 인증서 신뢰등급
    - 소개자로부터 인증서를 수신하는 경우, 인증된 개체에 대한 인증서를 저장하고 해당 인증서에 대한 신뢰등급을 부여함
    - 일반적으로 인증서의 신뢰등급은 소개자의 신뢰등급과 동일하며, 하나의 개체에 대해 여러 인증서가 저장될 수 있음

PGP

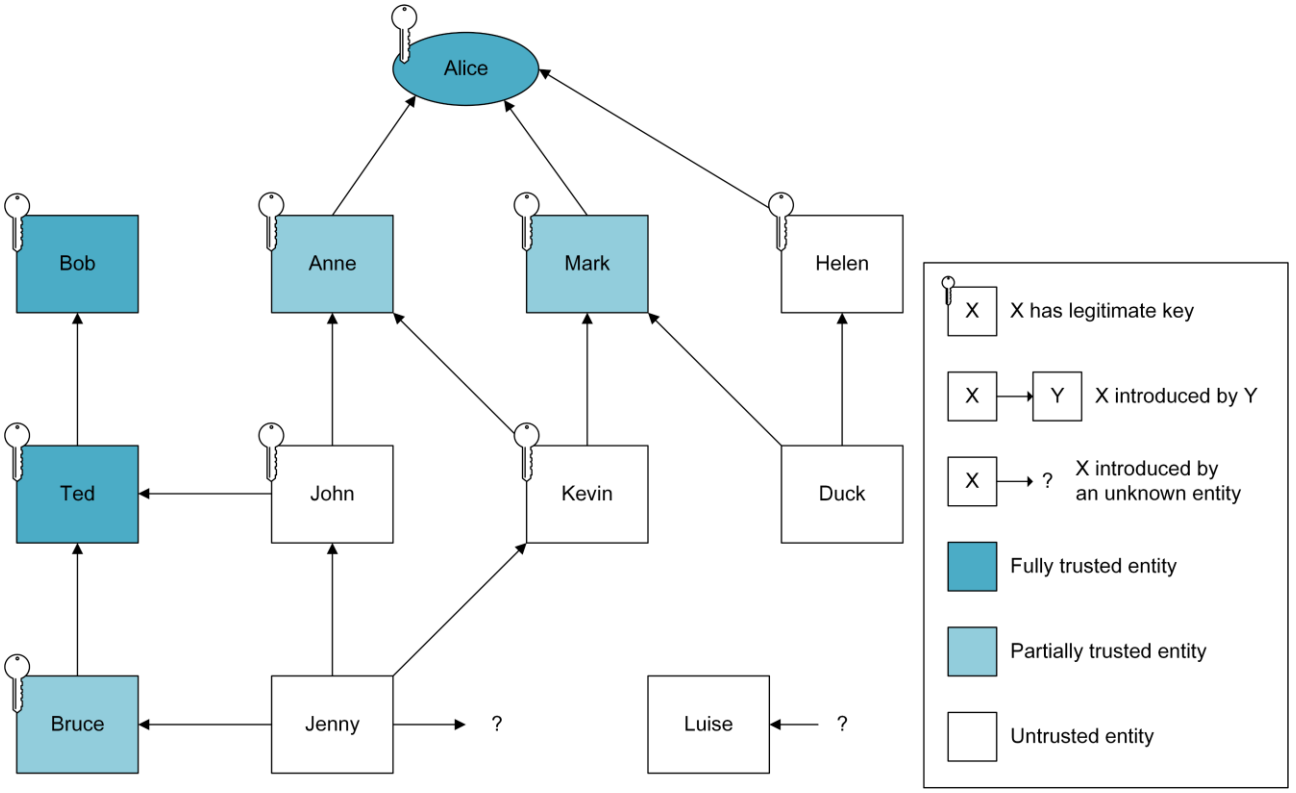
- 인증서(Certificate) (3/6)
  - 키 적법성 (1/2)
    - 사용자는 특정 개체(Entity)에 대한 인증서들의 신뢰등급을 종합하여 해당 개체에 대한 키 적법성 등급을 책정함
    - 인증서 신뢰등급별로 중요도를 부여하여 키 적법성 등급 책정에 활용할 수 있음
      - e.g., None 등급에 중요도 0, Partial 등급에 중요도 0.5, Full 등급에 중요도 1을 부여한 후, 중요도 총합이 1을 넘으면 적법한 것으로 봄
    - 특정 개체에 대한 키 적법성 등급에 따라 해당 개체에 대한 공개키 및 인증서를 보관하고 있어도 해당 개체의 공개키를 사용하지 않을 수 있음

PGP

- 인증서(Certificate) (4/6)
  - 키 적법성 (2/2)
    - 인증서를 통해 충분한 키 적법성을 확보하기 어려운 경우, 다음과 같은 방법으로 키 적법성을 확보할 수 있음
      - 두 사용자가 만나 물리적으로 공개키를 교환함
      - 전화 음성 상에서 공개키를 교환함
      - 전자우편으로 공개키를 교환하고 해당 공개키로 생성한 핑거프린트(Fingerprint)를 비교하여 변조 여부를 확인함
      - 외부의 인증기관을 이용하여 인증된 공개키를 획득함

PGP

- 인증서(Certificate) (5/6)
  - 신뢰모델
    - PGP는 사용자 간의 신뢰를 통해 신뢰할 수 있는 웹(Web of Trust)을 구현하여 분산형 관리에 적합함

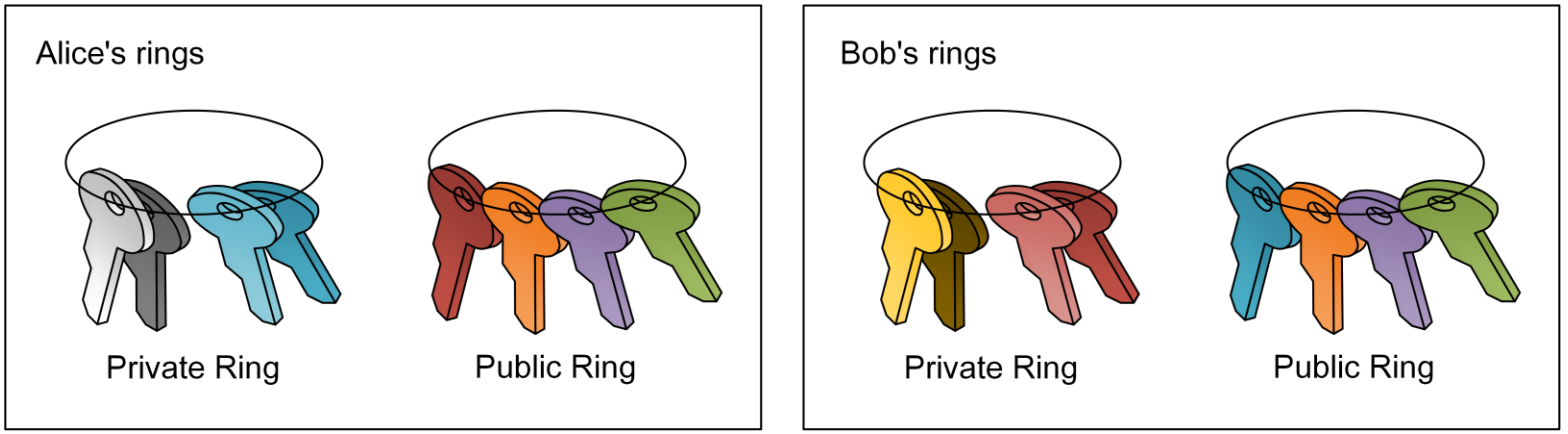


PGP

- 인증서(Certificate) (6/6)
  - 키 폐지
    - 키의 소유자는 키가 오래 되어 안전하지 않다고 판단하거나, 키 도난 등의 이유로 키가 손상되면 자신의 공개키를 폐지할 수 있음
    - 키의 소유자는 키를 폐지하기 위해서 자신이 서명한 폐지 인증서(Revocation Certificate)를 전송할 수 있음
    - 폐지 인증서는 반드시 폐지하려는 공개키에 대한 개인키로 서명되어야 하며, 공개 키 링에서 폐지하려는 공개키를 가지고 있는 모든 사람들에게 배포해야 함

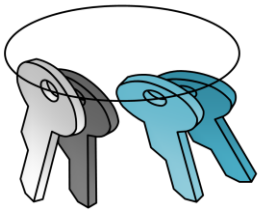
PGP

- 키 링(Key Ring) (1/12)
  - 개요
    - 전자우편을 송수신 할 때 필요한 키 등을 보관하기 위하여 키 링을 사용할 수 있음
    - 키 링은 자신의 공개키와 해당 공개키에 대한 개인키를 저장하는 개인 키 링(Private Key Ring)과, 다른 사용자의 공개키를 저장하는 공개 키 링(Public Key Ring)으로 분류됨



PGP

- 키 링(Key Ring) (2/12)
  - 키 링 테이블(Key Ring Table) (1/9)
    - 개인 키 링 (1/2)
      - User ID
        - 공개키-개인키 쌍을 식별하기 위한 값이며, 주로 사용자의 전자우편 주소임
      - Key ID
        - 공개키의 첫 8 Bytes로, 공개키를 빠르게 색인하기 위해 사용됨
      - Public key
        - 사용하는 공개키의 값이 저장됨
      - Encrypted private key
        - 공개키에 대응하는 개인키가 암호화되어 저장됨
      - Timestamp
        - 해당 키 쌍이 생성된 시각이 저장됨


Private Ring

User ID	Key ID	Public key	Encrypted private key	Timestamp
⋮	⋮	⋮	⋮	⋮

PGP

- 키 링(Key Ring) (3/12)
  - 키 링 테이블(Key Ring Table) (2/9)
    - 개인 키 링 (2/2)
      - 예시
        - Alice가 alice@anet.net과 alice@some.com이라는 두 개의 User ID를 가지고 있는 경우
        - 각 User ID에 대한 공개키-개인키 쌍을 하나씩 가지고 있음
        - 각 키에 대한 진법 표기, Timestamp의 날짜 표기 등은 구현에 따라 달라질 수 있음

User ID	Key ID	Public Key	Encrypted Private Key	Timestamp
alice@anet.net	AB13…45	AB13…45…59	32452398…23	031505-16:23
alice@some.com	FA23…12	FA23…12…22	564A4923…23	031504-08:11

PGP

- 키 링(Key Ring) (4/12)
  - 키 링 테이블(Key Ring Table) (3/9)
    - 공개 키 링 (1/7)
      - User ID
        - 공개키-개인키 쌍을 식별하기 위한 값이며, 주로 해당 개체의 전자우편 주소임
      - Key ID
        - 공개키의 첫 8 Bytes로, 공개키를 빠르게 색인하기 위해 사용됨
      - Public key
        - 해당 개체의 공개키 값이 저장됨
      - Producer trust
        - 해당 개체의 신뢰등급이 저장됨


Public Ring

User ID	Key ID	Public key	Producer trust	Certificate(s)	Certificate trust(s)	Key Legitimacy	Timestamp
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

PGP

- 키 링(Key Ring) (5/12)
  - 키 링 테이블(Key Ring Table) (4/9)
    - 공개 키 링 (2/7)
      - Certificate(s)
        - 하나 이상의 다른 소개자가 서명한 해당 개체의 인증서가 저장됨
      - Certificate trust(s)
        - 해당 인증서의 신뢰등급이 저장되며, 주로 소개자의 신뢰등급과 동일한 신뢰등급이 부여됨
      - Key Legitimacy
        - 인증서 신뢰등급을 종합하여 계산된 키 적법성 등급이 저장됨
      - Timestamp
        - 해당 개체에 대한 행이 생성된 시각이 저장됨


Public Ring

User ID	Key ID	Public key	Producer trust	Certificate(s)	Certificate trust(s)	Key Legitimacy	Timestamp
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

PGP

- 키 링(Key Ring) (6/12)
  - 키 링 테이블(Key Ring Table) (5/9)
    - 공개 키 링 (3/7)
      - 예시 (1/5)
        - Alice가 처음 공개 키 링을 생성한 경우, Alice 자신을 완전하게 신뢰하고 있으며, 이에 따라 Producer trust와 Key legitimacy는 완전한(F) 등급을 가짐

User ID	Key ID	Public Key	Prod. trust	Certificate	Cert. trust	Key legit.	Timestamp
Alice…	AB…	AB……	F			F	……

        - Alice는 완전히 신뢰하는 Bob과 공개키를 물리적으로 교환하여 공개 키 링에 추가하고, Certificate 없이도 Key legitimacy가 완전한(F) 등급을 가짐

User ID	Key ID	Public Key	Prod. trust	Certificate	Cert. trust	Key legit.	Timestamp
Alice…	AB…	AB……	F			F	……
Bob…	12…	12……	F			F	……

PGP

- 키 링(Key Ring) (7/12)
  - 키 링 테이블(Key Ring Table) (6/9)
    - 공개 키 링 (4/7)
      - 예시 (2/5)
        - Alice는 완전히 신뢰하는 Ted에 대해, Bob으로부터 인증서를 받았으며, Bob의 신뢰등급에 의해 Certificate trust가 완전한(F) 신뢰등급을 가짐
        - Ted가 하나 이상의 완전한(F) Certificate trust를 가지고 있으므로, 해당 공개키에 대한 Key legitimacy가 F로 책정됨

User ID	Key ID	Public Key	Prod. trust	Certificate	Cert. trust	Key legit.	Timestamp
Alice…	AB…	AB……	F			F	……
Bob…	12…	12……	F			F	……
Ted…	48…	48……	F	Bob's	F	F	……

PGP

- 키 링(Key Ring) (8/12)
  - 키 링 테이블(Key Ring Table) (7/9)
    - 공개 키 링 (5/7)
      - 예시 (3/5)
        - Alice는 부분적으로 신뢰하는 Anne에 대해, Bob으로부터 인증서를 받았으며, Bob의 신뢰등급에 의해 Certificate trust가 완전한(F) 신뢰등급을 가짐
        - Anne가 하나 이상의 완전한(F) Certificate trust를 가지고 있으므로, 해당 공개키에 대한 Key legitimacy가 F로 책정됨

User ID	Key ID	Public Key	Prod. trust	Certificate	Cert. trust	Key legit.	Timestamp
Alice…	AB…	AB……	F			F	……
Bob…	12…	12……	F			F	……
Ted…	48…	48……	F	Bob's	F	F	……
Anne…	71…	71……	P	Bob's	F	F	……

PGP

- 키 링(Key Ring) (9/12)
  - 키 링 테이블(Key Ring Table) (8/9)
    - 공개 키 링 (6/7)
      - 예시 (4/5)
        - Alice는 신뢰하지 않는 John에 대해, Anne로부터 인증서를 받았으며, Anne의 신뢰등급에 의해 Certificate trust가 부분적인(P) 신뢰등급을 가짐
        - John은 하나의 부분적인(P) Certificate trust만을 가지고 있으므로, 해당 공개키에 대한 Key legitimacy가 P로 책정됨

User ID	Key ID	Public Key	Prod. trust	Certificate	Cert. trust	Key legit.	Timestamp
Alice…	AB…	AB……	F			F	……
Bob…	12…	12……	F			F	……
Ted…	48…	48……	F	Bob's	F	F	……
Anne…	71…	71……	P	Bob's	F	F	……
John…	31…	31……	N	Anne's	P	P	……

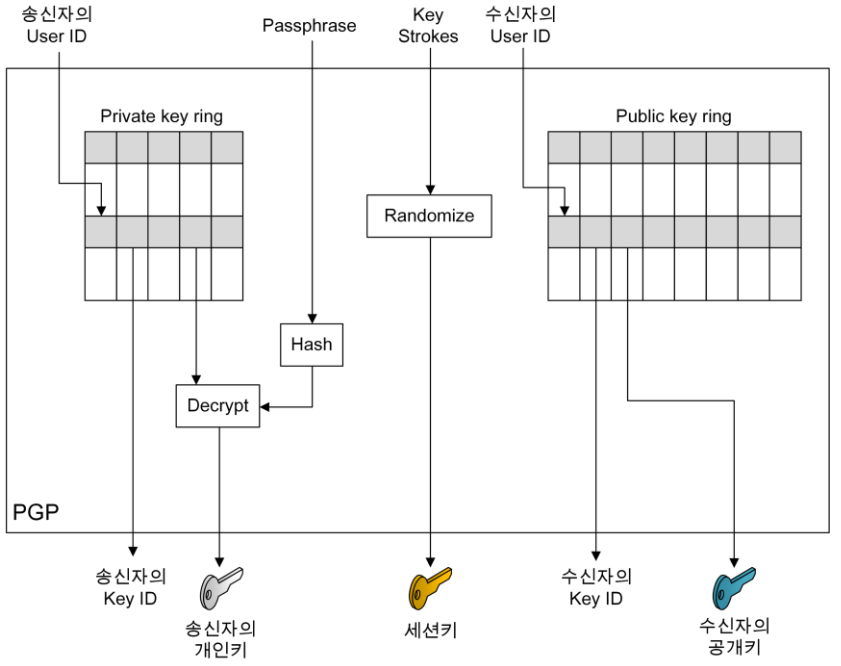
PGP

- 키 링(Key Ring) (10/12)
  - 키 링 테이블(Key Ring Table) (9/9)
    - 공개 키 링 (7/7)
      - 예시 (5/5)
        - Alice는 신뢰하지 않는 John에 대해, Ted로부터 인증서를 받았으며, Ted의 신뢰등급에 의해 Certificate trust가 완전한(F) 신뢰등급을 가짐
        - John은 하나의 부분적인(P) Certificate trust와, 하나의 완전한(F) Certificate trust를 가지고 있으므로, 해당 공개키에 대한 Key legitimacy가 F로 책정됨

User ID	Key ID	Public Key	Prod. trust	Certificate	Cert. trust	Key legit.	Timestamp
Alice…	AB…	AB……	F			F	……
Bob…	12…	12……	F			F	……
Ted…	48…	48……	F	Bob's	F	F	……
Anne…	71…	71……	P	Bob's	F	F	……
John…	31…	31……	N	Anne's Ted's	P F	F	……

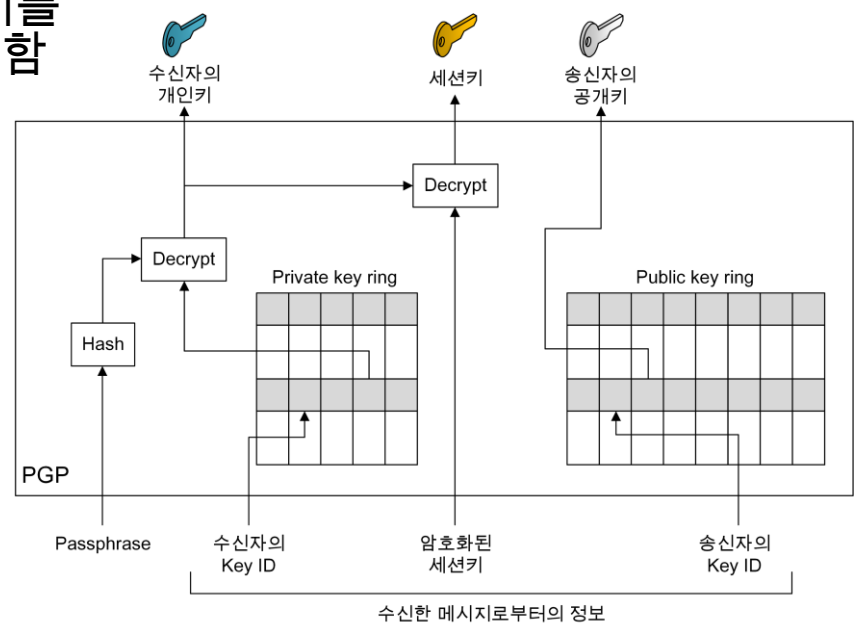
PGP

- 키 링(Key Ring) (11/12)
  - 정보 추출 (1/2)
    - 송신측
      - 전자우편을 전송하기 위해서는 송신자의 Key ID, 송신자의 개인키, 세션키, 수신자의 Key ID, 수신자의 공개키가 필요함
        - 개인 키 링에서 송신자의 User ID를 이용하여 송신자의 Key ID를 찾음
        - 개인 키 링에서 송신자의 User ID를 이용하여 송신자의 개인키를 찾고 Passphrase를 해시하여 얻은 키로 복호화함
        - 송신자의 Key Strokes를 시드로 난수 생성기를 이용하여 일회용 세션키를 생성함
        - 공개 키 링에서 수신자의 User ID를 이용하여 수신자의 Key ID와 공개키를 찾음



PGP

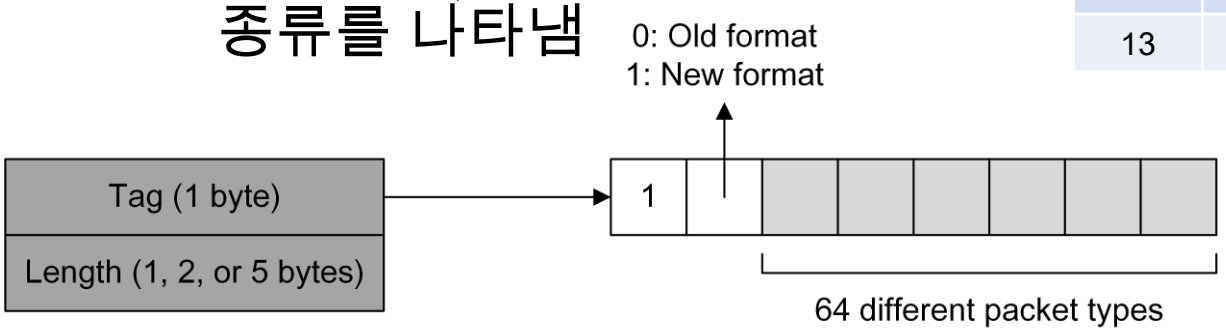
- 키 링(Key Ring) (12/12)
  - 정보 추출 (2/2)
    - 수신측
      - 수신받은 전자우편을 열람하기 위해 수신자의 개인키, 세션키, 송신자의 공개키가 필요함
        - 개인 키 링에서 수신자의 Key ID를 이용하여 수신자의 개인키를 찾고 Passphrase를 해시하여 얻은 키로 복호화함
        - 앞선 복호화로 얻은 수신자의 개인키를 이용하여 암호화된 세션키를 복호화함
        - 공개 키 링에서 송신자의 Key ID를 이용하여 송신자의 공개키를 찾음



PGP

- 패킷 (1/11)
  - 개요
    - PGP에서 메시지는 하나 또는 그 이상의 패킷으로 구성됨
    - 패킷은 공통 헤더(Generic Header)와 Body로 구분됨
  - 공통 헤더 (1/3)
    - Tag
      - 1 Byte 크기의 필드로 구성되며, 첫 비트는 Tag임을 나타내는 1 bit로 고정됨
      - 두 번째 비트는 헤더의 형식을 나타내며, 이후 비트는 패킷의 종류를 나타냄

Value	Packet type
1	Session key packet encrypted using a public key
2	Signature packet
5	Private-key packet
6	Public-key packet
8	Compressed data packet
9	Data packet encrypted with a secret key
11	Literal data packet
13	User ID packet



PGP

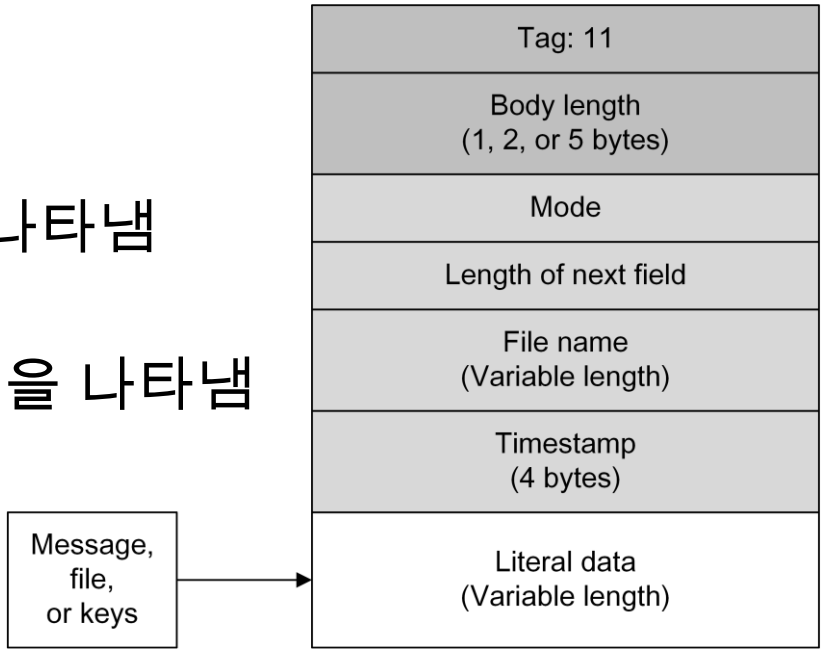
- 패킷 (2/11)
  - 공통 헤더 (2/3)
    - Length (1/2)
      - Length의 값이 192(11000000)보다 작은 경우, Length의 1 Byte가 나타내는 값이 Body의 길이 바이트 수가 됨
        - e.g., Length = 158이라면, Body Length = 158 Bytes
      - Length의 값이 192(11000000) ~ 223(11011111)인 경우, (첫 번째 바이트 $- 192) \times 256 +$ 두 번째 바이트 $+ 192$가 Body의 길이 바이트 수가 됨
        - e.g., Length = 200 100이라면, Body Length = $(200 - 9) \times 256 + 100 + 192 = 2340$ Bytes
      - Length의 값이 255(11111111)인 경우, 이후 4 Bytes로 표현된 값이 Body의 길이 바이트 수가 됨
        - e.g., Length = 255이고, 이후 4 Bytes가 39756이라면, Body Length = 39756 Bytes

PGP

- 패킷 (3/11)
  - 공통 헤더 (3/3)
    - Length (2/2)
      - Length의 값이 224(11100000) ~ 254(11111110)인 경우, $2^{(\text{첫번째 바이트} \,\&\, 31)}$가 이후 Body의 길이가 되며, 해당 Body 이후 Length-Body 구조를 반복하여 사용 가능함
        - e.g., Length = 230 | Body = 64 Bytes 데이터 | Length = 232 | Body = 256 Bytes 데이터 | …
      - 마지막 Body임을 알리기 위해서는 Length의 값이 224 ~ 254가 아니어야 함
        - e.g., Length = 230 | Body = 64 Bytes 데이터 | Length = 50 | Body = 50 Bytes 데이터

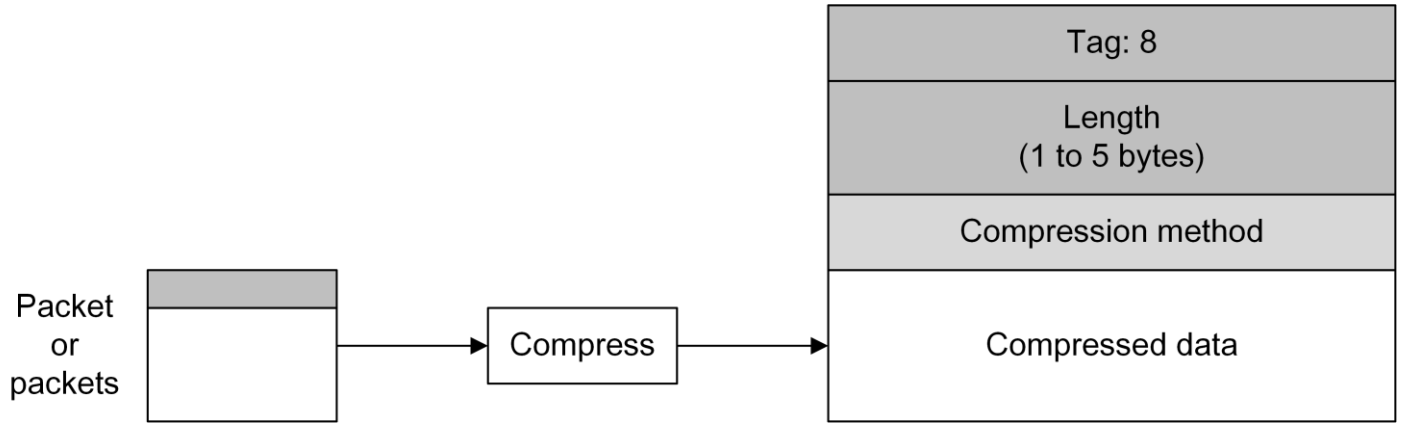
PGP

- 패킷 (4/11)
  - 문자 데이터 패킷(Literal Data Packet)
    - Mode
      - 데이터의 종류를 정의하는 1 Byte 필드로, 2진수인 경우 “b”, 텍스트인 경우 “t”를 ASCII 코드 값으로 나타냄
    - Length of next field
      - File name 필드의 길이를 나타냄
    - File name
      - 파일 이름을 ASCII 코드 문자열로 나타냄
    - Timestamp
      - 메시지 생성 혹은 마지막 수정 시간을 나타냄
    - Literal data
      - 2진수나 텍스트로 작성된 실제 데이터를 나타냄



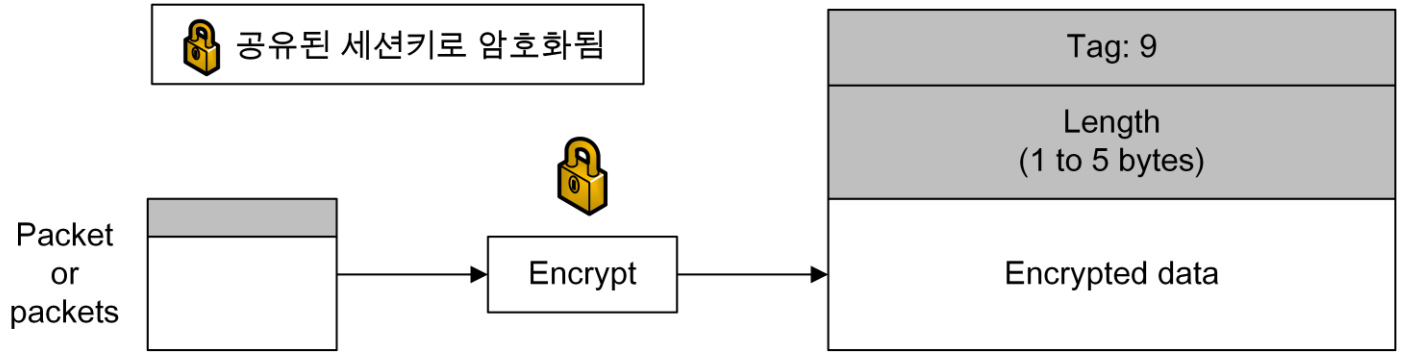
PGP

- 패킷 (5/11)
  - 압축 데이터 패킷(Compressed Data Packet)
    - Compression method
      - Compressed data의 압축에 사용된 압축 방법을 1 Byte로 나타냄
    - Compressed data
      - 메시지나 다른 패킷들을 Compression method의 방법으로 압축한 결과를 나타냄



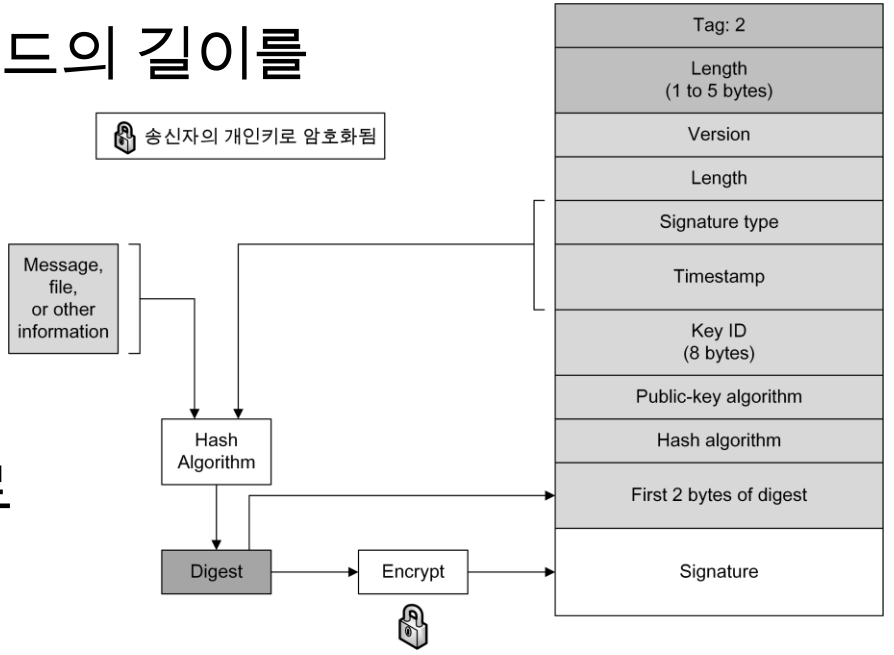
PGP

- 패킷 (6/11)
  - 암호화된 데이터 패킷(Encrypted Data Packet)
    - Encrypted data
      - 패킷들을 대칭키 알고리즘으로 암호화한 결과를 나타냄
      - 사용된 알고리즘은 Session Key Packet의 Encrypted session key 필드에서 나타냄



PGP

- 패킷 (7/11)
  - 서명 패킷(Signature Packet) (1/2)
    - Version
      - 사용된 PGP의 버전을 1 Byte로 나타냄
    - Length
      - Signature type과 Timestamp 필드의 길이를 나타냄
    - Signature type
      - 서명 목적, 서명 문서 종류를 1 Byte로 나타냄
    - Timestamp
      - 서명이 계산된 시간을 4 Bytes로 나타냄
    - Key ID
      - 서명 검증에 필요한 서명자의 공개키 ID를 8 Bytes로 나타냄

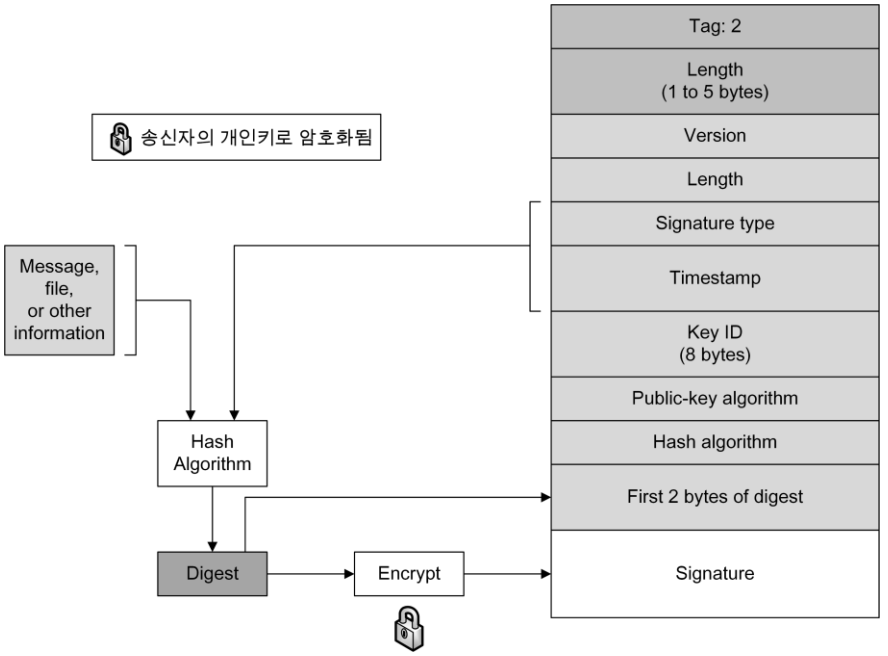


PGP

- 패킷 (8/11)
  - 서명 패킷(Signature Packet) (2/2)
    - Public-key algorithm
      - 다이제스트를 서명할 때 사용한 공개키 알고리즘을 1 Byte로 나타냄
    - Hash algorithm
      - 다이제스트를 생성할 때 사용한 해시 알고리즘을 1 Byte로 나타냄
    - First 2 bytes of digest
      - 빠른 무결성 검사를 위한 다이제스트의 첫 2 Bytes를 나타냄
    - Signature
      - 송신자가 생성한 서명을 나타냄

ID	설명
1	RSA (Encryption or signing)
2	RSA (For encryption only)
3	RSA (For signing only)
16	ElGamal (Encryption only)
17	DSS
18	ECDH
19	ECDSA
20	Reserved (Formerly ElGamal encryption or signing)
21	Reserved for Diffie-Hellman
100 ~ 110	Private or experimental use

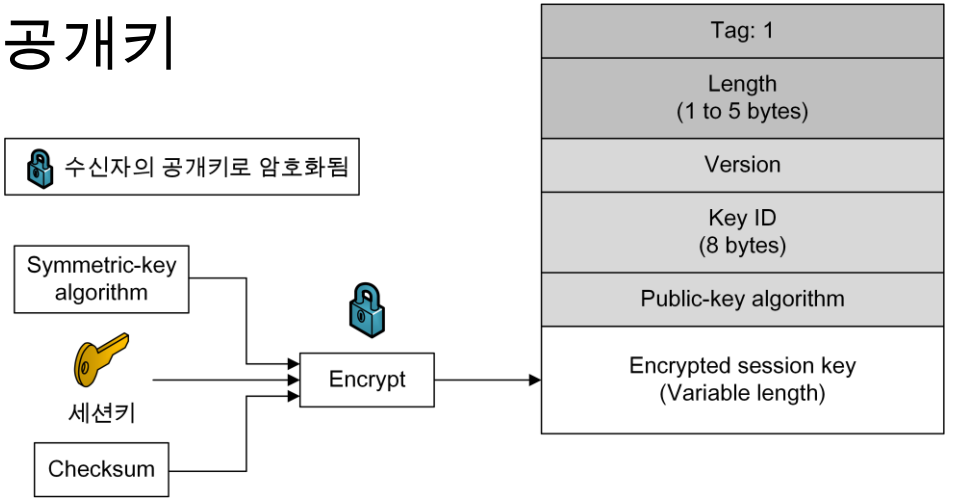
ID	설명
1	MD5
2	SHA-1
3	RIPEMD-160
8	SHA2-256
9	SHA2-384
10	SHA2-512
11	SHA2-224
12	SHA3-256
14	SHA3-512
100 ~ 110	Private or experimental use



PGP

- 패킷 (9/11)
  - 세션키 패킷(Session Key Packet)
    - Version
      - 사용된 PGP의 버전을 1 Byte로 나타냄
    - Key ID
      - 수신자의 공개키 ID를 8 Bytes로 나타냄
    - Public-key algorithm
      - 세션키를 암호화 할 때 사용한 공개키 알고리즘을 1 Byte로 나타냄
    - Encrypted session key
      - 1 Byte 대칭키 알고리즘 표기 + 세션키 + 세션키 각 바이트 합의 하위 2 Bytes를 수신자의 공개키로 암호화하여 나타냄

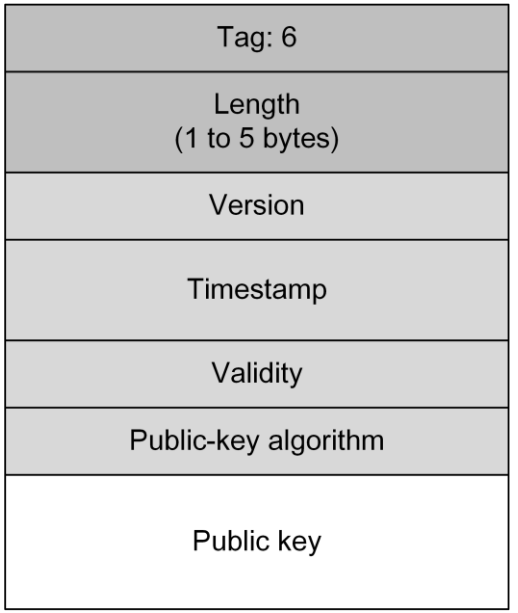
ID	설명
1	RSA (Encryption or signing)
2	RSA (For encryption only)
3	RSA (For signing only)
16	ElGamal (Encryption only)
17	DSS
18	ECDH
19	ECDSA
20	Reserved (Formerly ElGamal encryption or signing)
21	Reserved for Diffie-Hellman
100 ~ 110	Private or experimental use



PGP

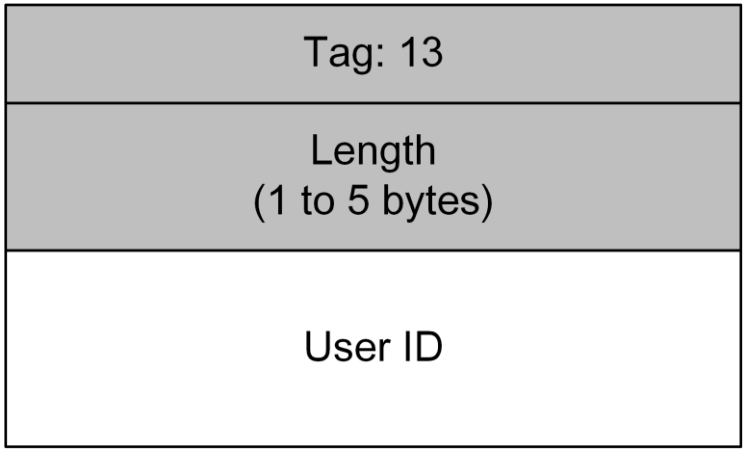
- 패킷 (10/11)
  - 공개키 패킷(Public-Key Packet)
    - Version
      - 사용된 PGP의 버전을 1 Byte로 나타냄
    - Timestamp
      - 공개키가 생성된 시간을 4 Bytes로 나타냄
    - Validity
      - 공개키의 유효시간을 2 Bytes로 나타냄
    - Public-key algorithm
      - 첨부한 공개키에 대한 공개키 알고리즘을 1 Byte로 나타냄
    - Public-key
      - 배포 및 전달할 공개키를 나타냄

ID	설명
1	RSA (Encryption or signing)
2	RSA (For encryption only)
3	RSA (For signing only)
16	ElGamal (Encryption only)
17	DSS
18	ECDH
19	ECDSA
20	Reserved (Formerly ElGamal encryption or signing)
21	Reserved for Diffie-Hellman
100 ~ 110	Private or experimental use



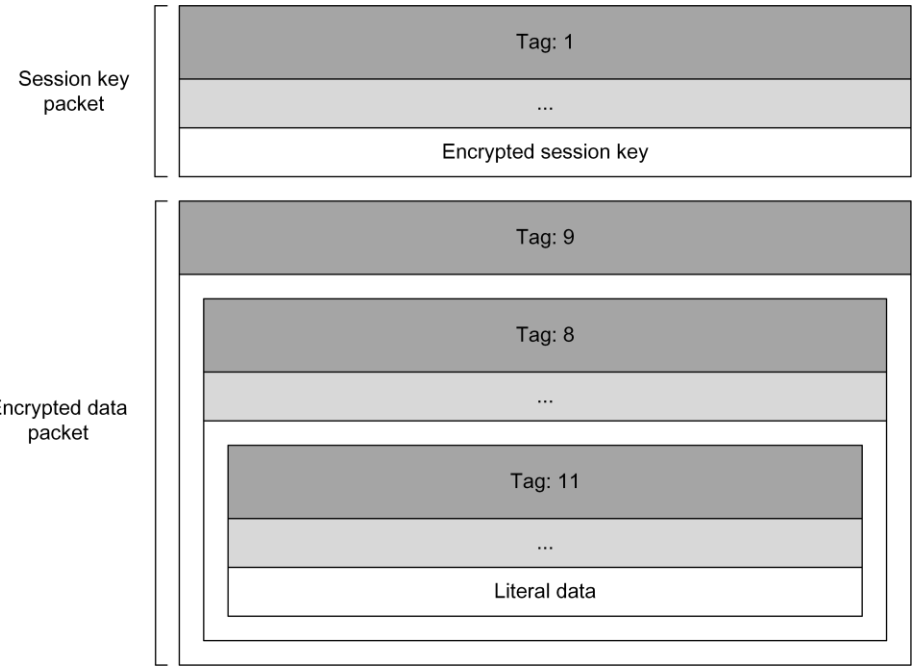
PGP

- 패킷 (11/11)
  - User ID 패킷(User ID Packet)
    - User ID
      - 송신자의 User ID를 나타내며, UTF-8로 부호화된 문자열로 저장됨
      - 주로 전자우편 주소에 따른 사용자의 이름으로 구성됨



PGP

- 메시지 (1/3)
  - 암호화된 메시지(Encrypted Message)
    - 메시지를 암호화하여 보낼 때 사용함
    - Session Key Packet과 해당 패킷의 세션키로 암호화된 Encrypted Data Packet으로 구성될 수 있음
      - Encrypted Data Packet의 Encrypted data는 Compressed Data Packet으로 구성됨
      - Compressed Data Packet의 Compressed data는 Literal Data Packet으로 구성됨
      - 추가적인 패킷을 전송하고자 하는 경우, 새로운 메시지로 생성해야 함



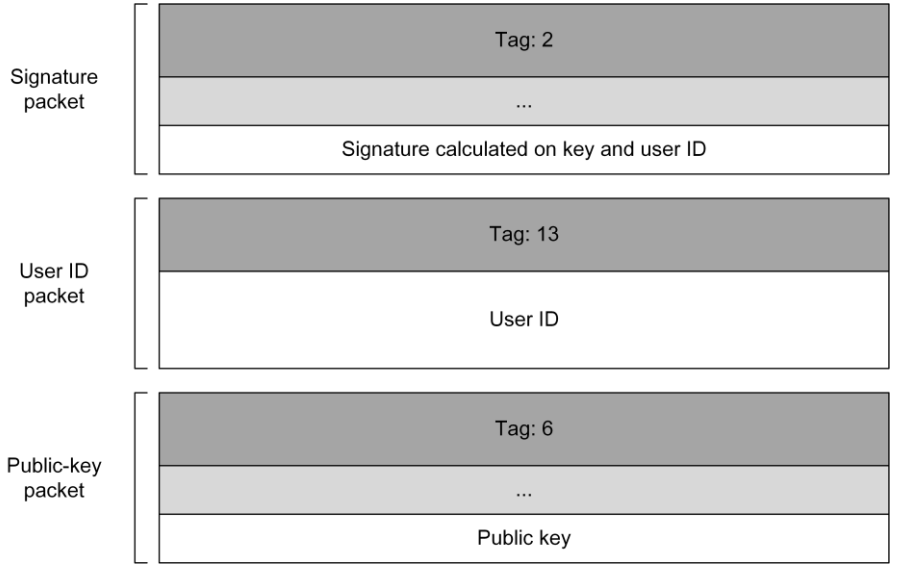
PGP

- 메시지 (2/3)
  - 서명된 메시지(Signed Message)
    - 메시지에 대한 서명을 보낼 때 사용함
    - Signature Packet과 Literal Data Packet으로 구성될 수 있음
      - Literal Data Packet은 필요에 따라 압축하여 Compressed Data Packet으로 만들거나, 압축한 후 암호화하여 Encrypted Data Packet으로 구성할 수 있음
      - 서명자가 여러 명인 경우, 서명자만큼의 Signature Packet을 추가할 수 있음



PGP

- 메시지 (3/3)
  - 인증서 메시지(Certificate Message)
    - 공개키가 해당 User ID를 가진 사용자의 공개키임을 서명으로 인증하여 보낼 때 사용함
    - Signature Packet, User ID Packet, Public-key Packet으로 구성될 수 있음
      - 인증서 구성에 따라 패킷이 추가될 수 있음



PGP

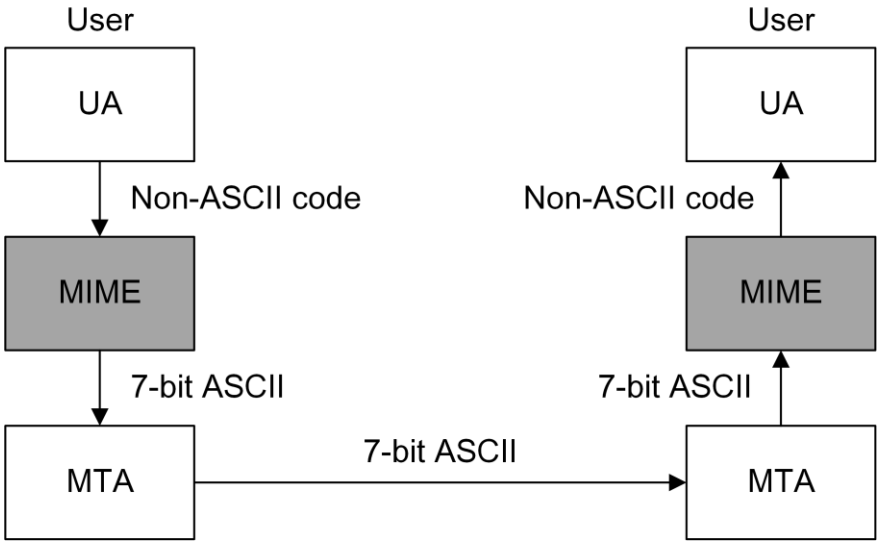
- 정리
  - PGP는 키 링을 이용하여 사용자 정보, 개인키, 공개키, 인증서 등의 중요 정보들을 관리함
  - PGP는 사용자 및 인증서에 신뢰등급을 부여하여 공개키의 적법성을 계산하고, 이를 통해 공개키의 사용 여부를 결정함
  - PGP는 데이터를 패킷 단위로 구분하고, 패킷들의 조합으로 메시지를 생성하여 전송함

목 차

- 보충
- 개요
- PGP
- S/MIME

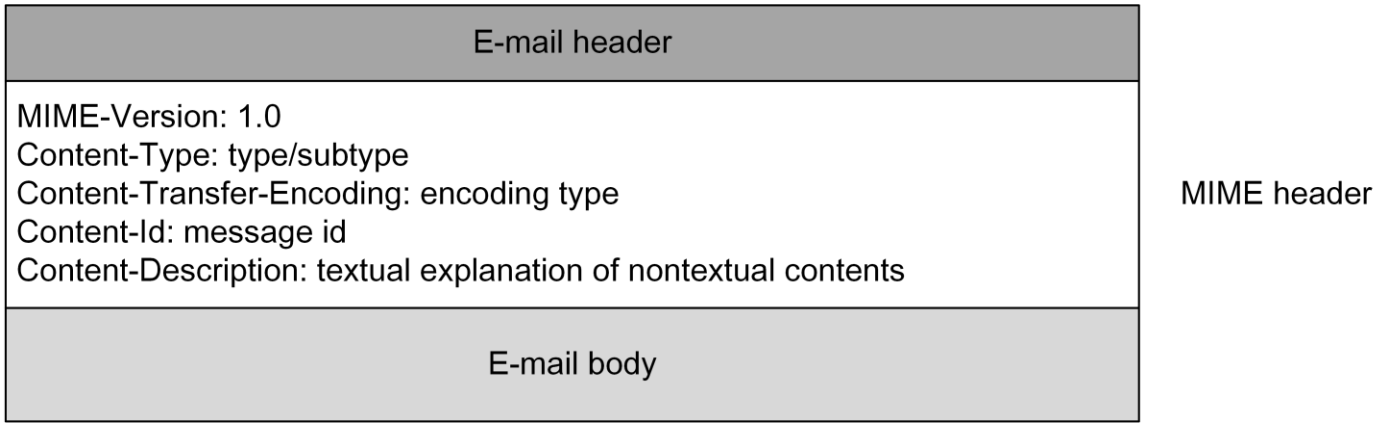
S/MIME

- MIME (Multipurpose Internet Mail Extensions) (1/14)
  - 개요 (1/2)
    - 전자우편 시스템에서 ASCII 형식으로 된 메시지만을 지원하는 경우, ASCII 코드로 표현하지 못하는 언어나 Binary 파일, 영상 및 음성 파일은 전송하지 못함
    - MIME 프로토콜은 Non-ASCII 데이터들을 7 Bits의 ASCII 코드로 변환하여 전송할 수 있음



S/MIME

- MIME (2/14)
  - 개요 (2/2)
    - MIME에서는 기존 전자우편 헤더에 추가할 수 있는 MIME Header를 사용함
  - MIME-Version
    - 사용된 MIME의 버전을 나타냄



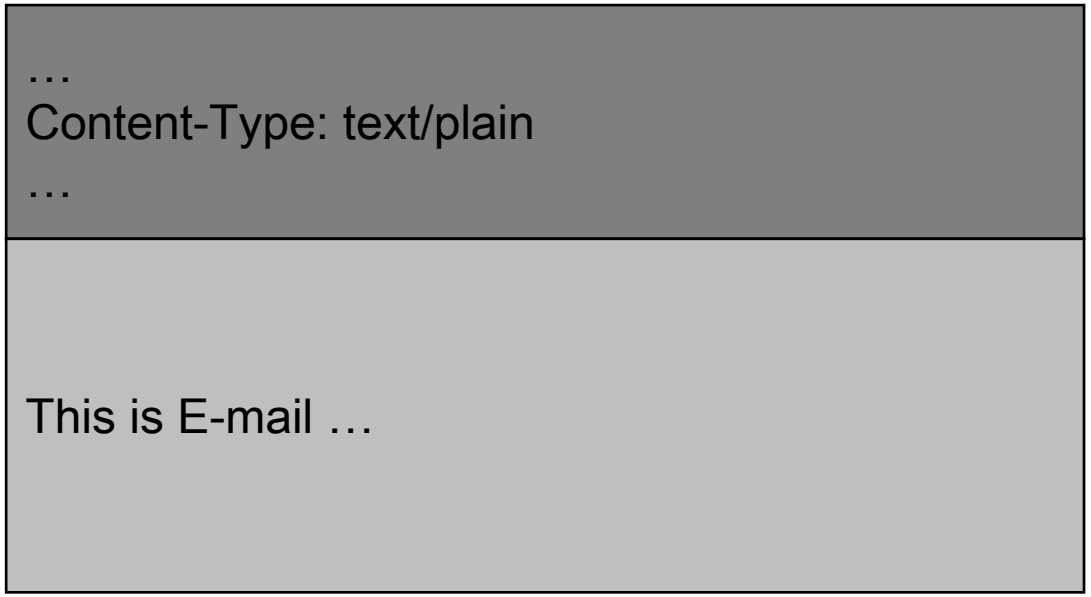
- MIME (3/14)
  - Content-Type (1/7)
    - 개요
      - 메시지의 종류를 “Content-Type : <type/subtype; parameters>” 형식으로 나타냄

RFC 822:
전자우편 메시지의 헤더와 본문 형식을 정의한 초기 인터넷 메시지 표준

Type	Subtype	Description
Text	Plain	Unformatted.
	HTML	HTML format.
Multipart	Mixed	Body contains ordered parts of different data types.
	Parallel	Same as above, but no order.
	Digest	Similar to Mixed, but the default is message/RFC822.
	Alternative	Parts are different versions of the same message.
Message	RFC822	Body is an encapsulated message.
	Partial	Body is a fragment of a bigger message.
	External-Body	Body is a reference to another message.
Image	JPEG	Image is in JPEG format.
	GIF	Image is in GIF format.
Video	MPEG	Video is in MPEG format.
Audio	Basic	Single channel encoding of voice at 8 KHz.
Application	PostScript	Adobe PostScript.
	Octet-stream	General binary data (eight-bit bytes).

S/MIME

- MIME (4/14)
  - Content-Type (2/7)
    - Text
      - Body가 7 Bits ASCII 코드로 구성됨
      - MIME에 의한 변환이 불필요함



S/MIME

- MIME (5/14)
  - Content-Type (3/7)
    - Multipart
      - Body가 여러 종류의 부분들로 구성됨
      - Mixed Subtype은 부분들이 수신자에게도 동일한 순서로 전달됨
      - Parallel Subtype은 부분들이 정렬되지 않은 채로 전달됨
      - Digest Subtype은 부분들의 구성이 Message/RFC822로 구성됨
      - Alternative Subtype은 같은 메시지가 서로 다른 형식을 사용하여 반복됨

```
…
Content-Type: multipart/XXXX; boundary=xxxx
…

--xxxx
Content-Type: text/plain;
…
--xxxx
Content-Type: image/gif;
…
--xxxx--
```

S/MIME

- MIME (6/14)
  - Content-Type (4/7)
    - Message (1/3)
      - Body가 다른 전자우편 메시지로 구성됨
      - RFC 822 Subtype은 단일 메시지 Body 전체로 구성됨

…
Content-Type: message/RFC822
…

From: alice@example.com
To: bob@example.com
Subject: Meeting Schedule
Date: Tue, 16 Sep 2026 10:00:00
…

S/MIME

- MIME (7/14)
  - Content-Type (5/7)
    - Message (2/3)
      - Partial Subtype은 파라미터를 통해 id와 단편화 번호를 알리고, 단편화된 전자우편 메시지의 일부로 구성됨

```
…
Content-Type: message/partial; id="alice@anet.net"; number=1; total=3;
…

From: alice@example.com
To: bob@example.com
Subject: Meeting Schedule
Date: Tue, 16 Sep 2026 10:00:00
…
```

S/MIME

- MIME (8/14)
  - Content-Type (6/7)
    - Message (3/3)
      - External-body Subtype은 파라미터를 통해 외부 데이터의 위치와 접근 방식을 알리고, 외부 데이터에 대한 설명으로 구성됨

…
Content-Type: message/external-body; name="report.txt"; site=anet.net;
access-type="ftp";
…

The file "report.txt" is stored on the FTP server fhda.edu.
Please retrieve the file using FTP.
…

S/MIME

- MIME (9/14)
  - Content-Type (7/7)
    - Image
      - Body가 이미지 파일로 구성됨
    - Video
      - Body가 비디오 파일로 구성됨
    - Audio
      - Body가 음성 파일로 구성됨
    - Application
      - Body가 앞선 Subtype으로 정의할 수 없는 데이터로 구성됨

S/MIME

- MIME (10/14)
  - Content-Transfer-Encoding (1/4)
    - 개요
      - 메시지의 부호화 방식을 “Content-Transfer-Encoding: <type>” 형식으로 나타냄

NVT ASCII (Network Virtual Terminal ASCII):
서로 다른 시스템 간에 텍스트를 일관된 형식으로 전송하기 위해 정의된 표준 ASCII 기반 네트워크 문자 표현 방식

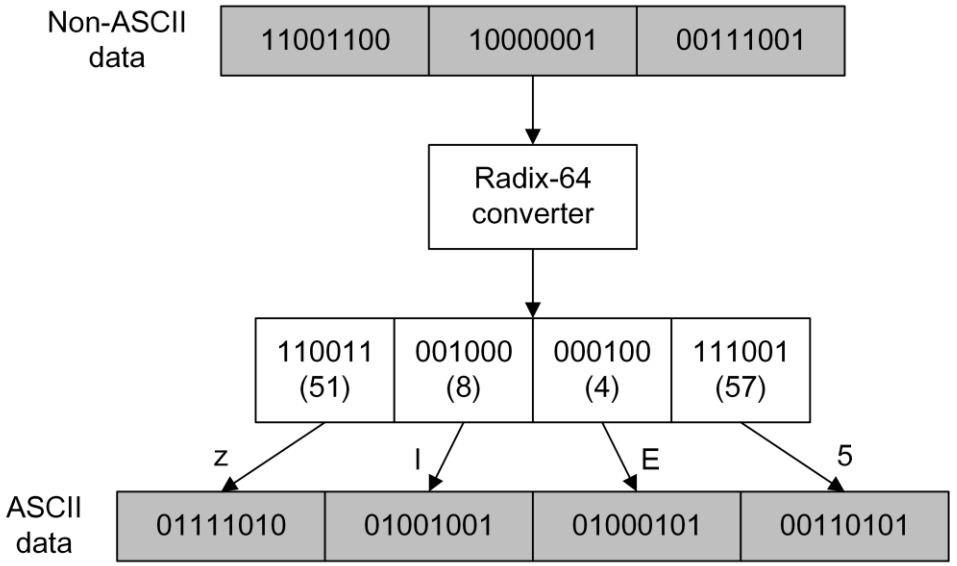
Type	Description
7bit	NVT ASCII characters and short lines.
8bit	Non-ASCII characters and short lines.
Binary	Non-ASCII characters with unlimited-length lines.
Radix-64	6-bit blocks of data are encoded into 8-bit ASCII characters using Radix-64 conversion.
Quoted-printable	Non-ASCII characters are encoded as an equal sign followed by an ASCII code.

S/MIME

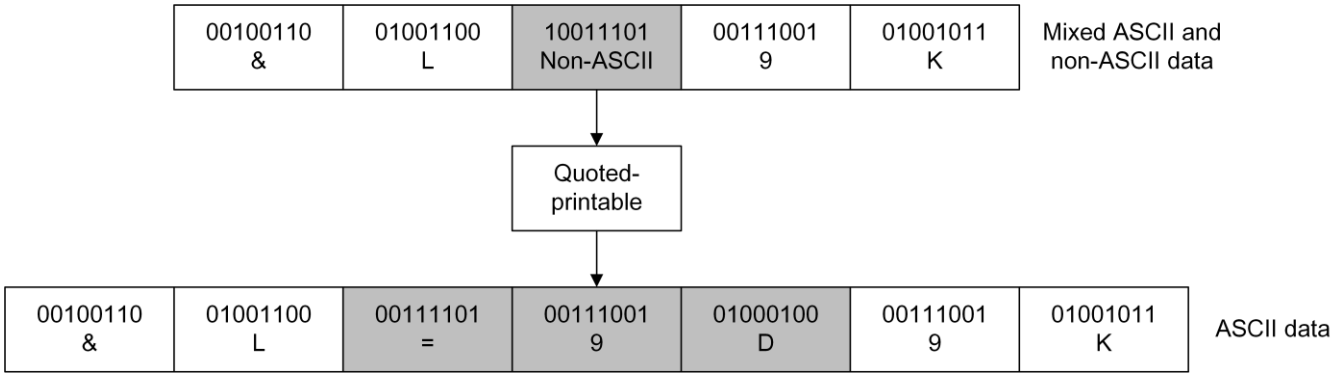
- MIME (11/14)
  - Content-Transfer-Encoding (2/4)
    - 7bit
      - 7-bit NVT ASCII 부호화를 사용함
      - 한 줄의 최대 길이는 1,000 글자를 초과하지 못함
    - 8bit
      - 8-bit 부호화를 사용함
      - 한 줄의 최대 길이는 1,000 글자를 초과하지 못함
      - ASCII가 아닌 문자를 전송할 수 있으나, 하위 SMTP 프로토콜이 8 Bits Non-ASCII 전송이 가능해야 함
    - Binary
      - 8-bit 부호화를 사용함
      - 한 줄의 최대 길이는 1,000 글자를 초과하지 못함
      - ASCII가 아닌 문자를 전송할 수 있으나, 하위 SMTP 프로토콜이 2진 데이터를 전송할 수 있어야 함

S/MIME

- MIME (12/14)
  - Content-Transfer-Encoding (3/4)
    - Radix-64
      - 8-bit Non-ASCII 데이터를 7-bit ASCII 데이터로 변환함
      - 2진 데이터를 24-bit 블록으로 분할하고, 해당 블록을 다시 6-bit 단락(Section) 4개로 분할하여, 각 단락을 64개의 8-bit ASCII 문자 중 하나로 변환함



- MIME (13/14)
  - Content-Transfer-Encoding (4/4)
    - Quoted-printable
      - 대부분의 ASCII 데이터는 그대로 전송하고, Non-ASCII 데이터만 ASCII 데이터로 변환함
      - 변환이 필요한 1바이트 데이터를 등호(=)와 해당 바이트의 16진수 값 2자리로 표현하여, 3개의 ASCII 문자로 변환함



S/MIME

- MIME (14/14)
  - Content-Id
    - 다중 메시지 상황에서 전체 메시지를 유일하게 식별함
  - Content-Description
    - Body에 대해 사용자가 읽을 수 있는 형태로 설명함
- S/MIME로의 확장
  - 기존 MIME에는 보안이 적용된 콘텐츠를 나타내는 Subtype이 존재하지 않음
  - MIME 구조를 유지하며 S/MIME (Secure/MIME)을 구현하기 위해, Application Content-Type의 Subtype으로 ‘pkcs7-mime’를 추가하여 사용함

S/MIME

- CMS (Cryptographic Message Syntax) (1/5)
  - 개요
    - application/pkcs7-mime Subtype을 통해 CMS 콘텐츠 유형을 표시함

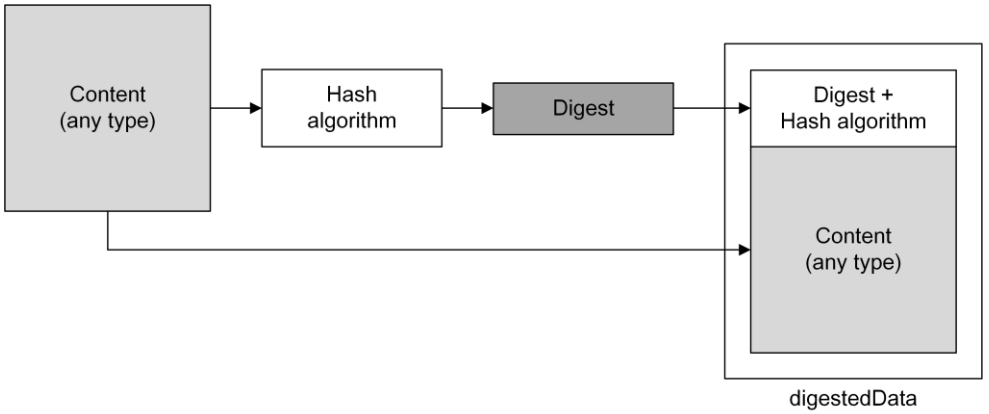
```
…
Content-Type: application/pkcs7-mime; mime-type=enveloped-data
Content-Transfer-Encoding: Radix-64
...
Content-Description: attachment

name="report.txt";
cb32ut67f4bhijHU21oi87eryb0287hmnklsgFDoY8bc659GhIGfH654
3mhjkdsaH23YjBnmNybmlkzjhgfdyhGe23Kjk34XiuD678Es16se09j
y76jHuytTMDcbnmlkjgfFdiuyu678543m0n3hG34un12P2454Hoi87e
2ryb0H2MjN6KuyrlsgFDoY897fk923jljk1301XiuD6gh78EsUyT23y
```

CMS 유형	설명
Data Content Type	임의의 데이터를 그대로 포함하는 기본 CMS 콘텐츠 유형
Digested-Data Content Type	콘텐츠와 해시값을 함께 포함하여 데이터 변경 여부를 확인함
Signed-Data Content Type	콘텐츠에 전자서명을 적용하여 무결성, 송신자 인증 등을 제공함
Authenticated-Data Content Type	MAC을 이용하여 콘텐츠의 무결성과 인증을 제공함
Encrypted-Data Content Type	콘텐츠를 대칭키로 암호화하지만, 해당 암호키를 전달하기 위한 수신자 정보는 포함하지 않음
Enveloped-Data Content Type	콘텐츠를 대칭키로 암호화하고, 해당 키를 각 수신자가 얻을 수 있도록 수신자 정보를 함께 포함

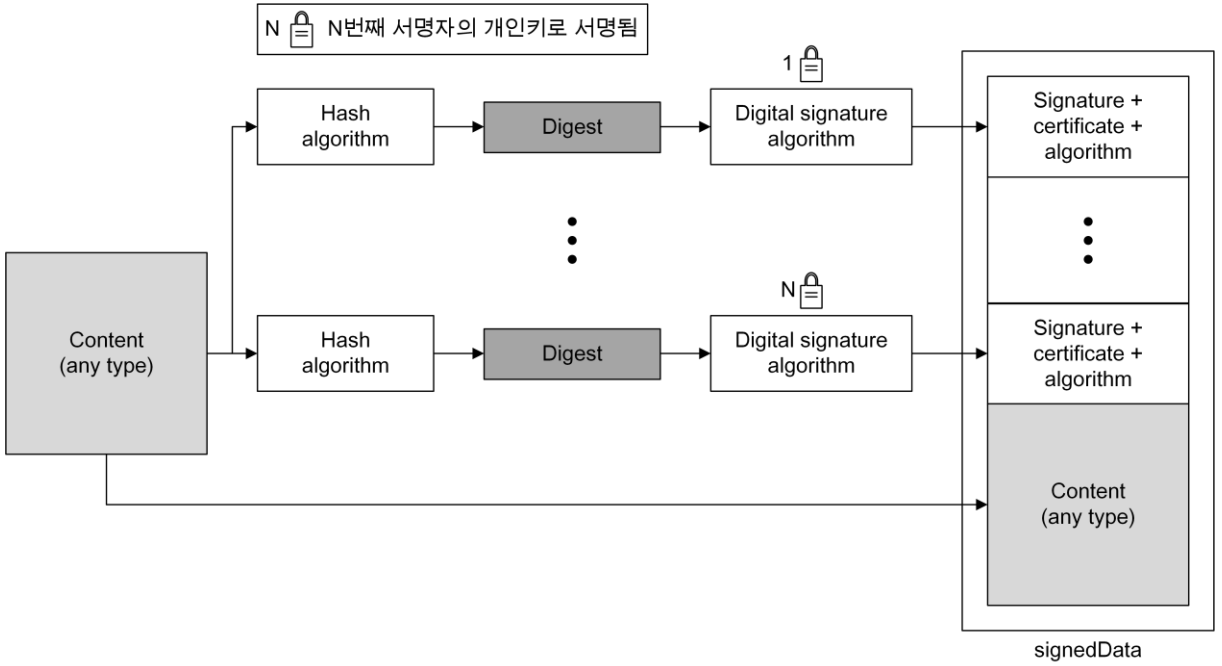
S/MIME

- CMS (Cryptographic Message Syntax) (2/5)
  - Data Content Type
    - 임의의 문자열(Arbitrary String)이며, 생성된 객체(Object)는 Data라고 불림
  - Digested-Data Content Type
    - 임의의 유형과 해당 콘텐츠의 Digest가 포함되며, 생성된 객체는 digestedData라고 불림



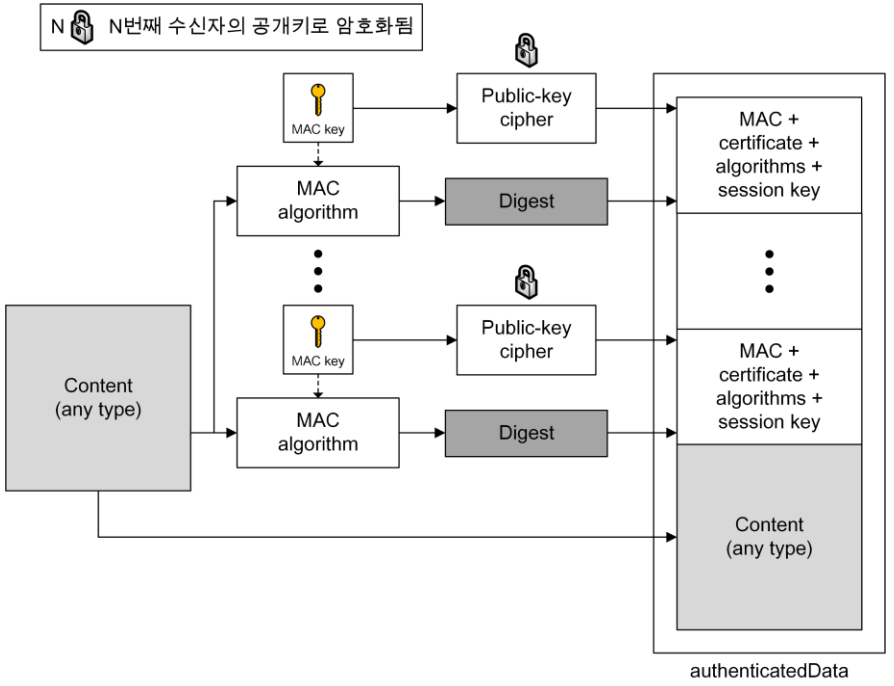
S/MIME

- CMS (Cryptographic Message Syntax) (3/5)
  - Signed-Data Content Type
    - 임의의 유형과 서명 값(Signature Value)을 포함하며, 생성된 객체는 signedData라고 불림



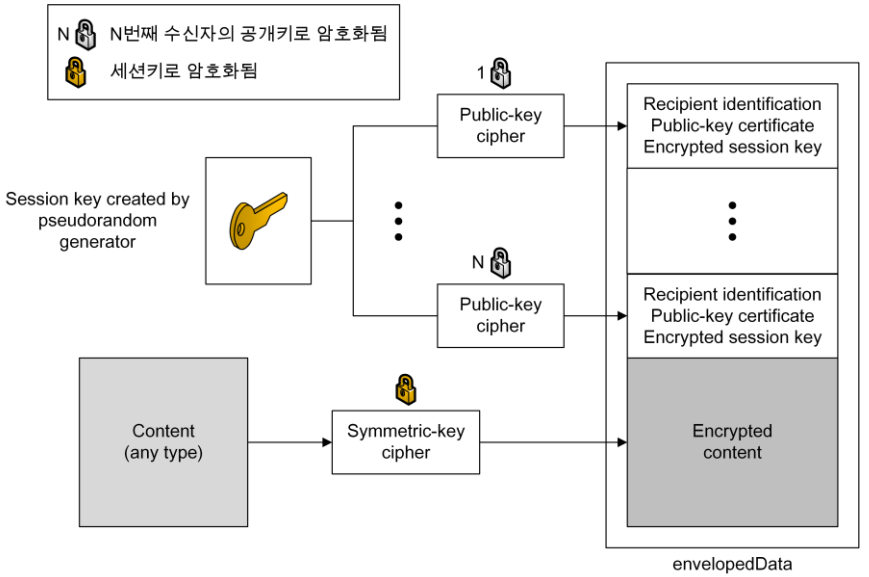
S/MIME

- CMS (Cryptographic Message Syntax) (4/5)
  - Authenticated-Data Content Type
    - 임의의 유형과 MAC (Message Authentication Code)가 포함되며, 생성된 객체는 authenticatedData라고 불림



S/MIME

- CMS (Cryptographic Message Syntax) (5/5)
  - Encrypted-Data Content Type
    - 암호화된 임의의 유형만 포함되며, 생성된 객체는 encryptedData라고 불림
  - Enveloped-Data Content Type
    - 임의의 유형과 암호화된 키 및 인증서가 포함되며, 생성된 객체는 envelopedData라고 불림



S/MIME

- 키 관리
  - 공개키는 CA로부터 발급받은 X.509를 인증서로 사용함
  - 사용자는 CA의 신뢰성을 지속적으로 검증해야 함
- 암호 알고리즘
  - “must support”는 반드시 지원해야 하는 알고리즘을 의미함
  - “should support”는 지원을 권고하는 알고리즘을 의미함

Algorithm	Sender must support	Receive must support	Sender should support	Receiver should support
Content-encryption algorithm	AES-128-GCM, AES-256-GCM	AES-128-GCM, AES-256-GCM	ChaCha20-Poly1305	ChaCha20-Poly1305
Session-key encryption algorithm	ECDH P-256, X25519 + HKDF-SHA-256	ECDH P-256, X25519 + HKDF-SHA-256	RSAES-OAEP	RSAES-OAEP
Hash algorithm	SHA-256, SHA-512	SHA-256, SHA-512	-	-
Digest-encryption algorithm	ECDSA P-256. EdDSA(Ed25519) 중 하나 이상	ECDSA P-256, EdDSA(Ed25519) 모두	RSASSA-PSS + SHA-256	RSASSA-PSS + SHA-256
Message-authentication algorithm	-	-	AuthEnvelopedData에서 인증 암호 사용	AuthEnvelopedData에서 인증 암호 사용

S/MIME

- 정리
  - MIME는 MIME 헤더를 이용하여 전자우편의 메시지가 어떠한 유형의 데이터인지 나타냄
  - S/MIME는 MIME에 보안용 Subtype을 추가하여 보안이 적용된 콘텐츠를 전송함
  - 공개키는 CA로부터 X.509 인증서를 발급받아 사용하여 중앙집중식 관리에 적합함

Thanks!

이 성 현(seonghyeon@pel.sejong.ac.kr)

부록 #1

- PGP 사용 알고리즘 (1/4)
  - 공개키 알고리즘
    - 다이제스트 서명 및 세션키 암호화에 사용됨

ID	설명
1	RSA (Encryption or signing)
2	RSA (For encryption only)
3	RSA (For signing only)
16	ElGamal (Encryption only)
17	DSS
18	ECDH
19	ECDSA
20	Reserved (Formerly ElGamal encryption or signing)
21	Reserved for Diffie-Hellman
100 ~ 110	Private or experimental use

부록 #1

- PGP 사용 알고리즘 (2/4)
  - 대칭키 알고리즘
    - 메시지 데이터 암호화에 사용됨

ID	설명
0	No encryption / Plaintext
1	IDEA
2	Triple DES
3	CAST5-128
4	Blowfish
7	AES-128
8	AES-192
9	AES-256
100 ~ 110	Private or experimental use

부록 #1

- PGP 사용 알고리즘 (3/4)
  - 해시 알고리즘
    - 메시지 다이제스트 생성에 사용됨

ID	설명
1	MD5
2	SHA-1
3	RIPEMD-160
8	SHA2-256
9	SHA2-384
10	SHA2-512
11	SHA2-224
12	SHA3-256
14	SHA3-512
100 ~ 110	Private or experimental use

부록 #1

- PGP 사용 알고리즘 (4/4)
  - 압축 알고리즘
    - 메시지 데이터 압축에 사용됨

ID	설명
0	Uncompressed
1	ZIP
2	ZLIB
3	BZip2
100 ~ 110	Private or experimental use

부록 #2

- Signature Type
  - 0x00과 0x01은 서명 대상 파일 종류를 나타냄
  - 0x10 ~ 0x13은 User ID 및 공개키에 대한 서명 수준을 나타냄
  - 0x30은 키 폐지를 위한 폐지 인증서임을 나타냄

Value	Signature
0x00	Signature of a binary document (message or file).
0x01	Signature of a text document (message or file).
0x10	Generic certificate of a User ID and public-key packet. The signer does not make any particular assertion about the owner of the key.
0x11	Personal certificate of a User ID and public-key packet. No verification is done on the owner of the key.
0x12	Casual certificate of a User ID and public-key packet. Some casual verification done on the owner of the key.
0x13	Positive certificate of a User ID and public-key packet. Substantial verification done.
0x30	Certificate revocation signature. This removes an earlier certificate(0x10 through 0x13).

부록 #3

- Radix-64 부호화 표

Value	Code	Value	Code	Value	Code	Value	Code	Value	Code	Value	Code
0	A	11	L	22	W	33	h	44	s	55	3
1	B	12	M	23	X	34	i	45	t	56	4
2	C	13	N	24	Y	35	j	46	u	57	5
3	D	14	O	25	Z	36	k	47	v	58	6
4	E	15	P	26	a	37	l	48	w	59	7
5	F	16	Q	27	b	38	m	49	x	60	8
6	G	17	R	28	c	39	n	50	y	61	9
7	H	18	S	29	d	40	o	51	z	62	+
8	I	19	T	30	e	41	p	52	0	63	/
9	J	20	U	31	f	42	q	53	1		
10	K	21	V	32	g	43	r	54	2		